NASA Contractor Report 3954

Flow Prediction for Propfan Engine Installation Effects on Transport Aircraft at Transonic Speeds

S. S. Samant and N. J. Yu

CONTRACT NAS1-17250
JANUARY 1986

NASA

NASA Contractor Report 3954

Flow Prediction for Propfan Engine Installation Effects on Transport Aircraft at Transonic Speeds

S. S. Samant and N. J. Yu
Boeing Commercial Airplane Company
Seattle, Washington

Prepared for
Langley Research Center
under Contract NAS1-17250

National Aeronautics
and Space Administration

**Scientific and Technical
Information Branch**

1986

TABLE OF CONTENTS

TABLES

FIGURES

1.0 SUMMARY

An Euler-based method for aerodynamic analysis of turboprop transport aircraft at transonic speeds has been developed. In this method, inviscid Euler equations are solved over surface-fitted grids constructed about aircraft configurations. Propeller effects are simulated by specifying sources of momentum and energy on an actuator disk located in place of the propeller. A preliminary version of an approach to embed the exhaust plume within the global Euler solution has also been developed for more accurate treatment of the exhaust flow. The resulting system of programs is capable of handling wing-body-nacelle-propeller configurations. The propeller disks may be tractors or pushers and may represent single or counterrotation propellers. Results from analyses of three test cases of interest (a wing alone, a wing-body-nacelle model, and a wing-nacelle model) are presented. A user's manual for executing the system of computer programs including formats of various input files, sample job decks, and sample input files is provided in appendices.

2.0 INTRODUCTION

Several studies of advanced propeller airplanes such as those summarized in Reference 1 show that high speed propfan propulsion systems can be more efficient than their turbofan counterparts at cruise speeds of up to about Mach 0.8. It is essential, however, to integrate the propulsion system properly with the configuration if the efficiency advantages are to be realized in actual flight. This is especially true of wing-mounted tractor installations on supercritical wings since the blockage, slipstream swirl, and added total pressure could deteriorate the already supercritical flow over the wing resulting in large interference drag penalties. Experimental studies by Welge and others[2,3] and by Smith and Levin[4,5] show clearly that the size and shape of the nacelle can have very large effects on the aerodynamics of the wing. This is confirmed by computational studies[6] indicating that proper nacelle-wing integration can eliminate or significantly reduce the adverse effects.

A method for properly integrating a propfan engine must deal correctly with the complexity of the flow. Experimental methods may be used to a degree, but at present, computational methods are best suited to provide the detailed knowledge of the flow essential to arrive at a superior design in a cost effective manner. Any such method must address such important issues as: the complex geometry, the nonlinearities associated with the transonic and highly rotational flow caused by the swirl, the increased total pressure in the flow downstream of the propfan, and the effects of the boundary layer and the exhaust plume. Such complex flow is beyond the scope of approximations underlying the potential method used in Reference 6. Thus, a better choice for such problems would be to solve the Euler equations.

In this report, one such method will be presented. Solving for aerodynamic characteristics of flows about complex geometries involves two major tasks. Firstly, it is necessary to construct a well distributed grid about the configuration. Secondly, the Euler equations are to be solved over such a grid using a suitable numerical method. In the method to be described in this report, the grid is generated using a technique developed originally by Thompson[8] and later adopted by Yu[9,10] for more complex geometries. The inviscid Euler equations are solved using a method developed originally by Jameson et al[7] and the propeller is simulated as an actuator disk lying in one of the grid planes where appropriate boundary conditions are prescribed to simulate the power loading of the propeller. Both tractor and pusher types of arrangements can be analyzed using this method. Since the primary objective of the present method is to calculate the propeller power interference effects on airplane aerodynamics, this simplified propeller representation is considered to be adequate.

A more refined version of this method that uses local grid embedding has been developed. Local grid embedding allows for greater geometric detail in the vicinity of the nacelle so that the details of the exhaust plume and its interaction with the surrounding flow may be calculated. Numerical issues associated with grid embedding methodology such as proper flux formulation and treatment of grid kinks across the common boundary of the embedded and the global regions are also discussed in general. This technique has been applied to the problem of an interacting exhaust plume. Further development of this method for more general application is recommended.

Computed results for three test cases are presented to demonstrate the capabilities of the method. The first test case includes a NASA-Ames wing-body-nacelle-propeller model including preliminary results for the grid embedding method of simulating the exhaust plume. The second test case includes the Onera M6 wing. The third test case includes the NASA-Langley wing-nacelle-propeller model.

The user's manual for the various computer programs is presented as an appendix. The system execution procedures and formats for the input files are given there. Sample job control decks and input files are also given in appendices.

3.0 NOMENCLATURE

a	Speed of sound
A,B,C	Coefficients in a typical 2D integral boundary layer equation
C_p	Pressure coefficient
E	Energy
f	Vector defined in Equation 2
F_x,F_y,F_z	Propeller force distribution, Equation 5
g	Vector defined in Equation 2
H	Total enthalpy
m	Mass flux used in boundary layer solution, Equation 13
M	Mach number
n	Unit normal vector
p	Pressure
q	Velocity vector
S	Cell face area, Equation 5g
S_x,S_y,S_z	Components of the cell face area in the coordinate directions
t	Time
T	Temperature
u,v,w	Physical velocity components in x,y,z directions
u_i	Inviscid version of the tangential velocity at the edge of the boundary layer
u_s	Tangential velocity at the edge of the boundary layer
U	Flux velocity, Equation 5f
W	Flow variable vector in Equation 2
x,y,z	Physical coordinates
x/c	Nondimensionalized chord station

α	Angle of attack
δ	Displacement thickness
$\delta(r)$	Swirl angle
η	Nondimensionalized span station, semispan fraction
γ	Ratio of specific heats
ϱ	Density
σ	Boundary layer mass source, Equation 11
ω	Relaxation parameter, Equation 13

Subscripts

o	Value to be used in outer inviscid calculations
1	Upstream of the disk
2	Downstream of the disk
n	Normal to the surface
s	Direction tangent to the surface
t	Total value
x,y,z	Coordinate direction
∞	At infinity

4.0 GRID GENERATION

The primary input to the flow solution algorithm is a reasonably smooth volume grid. In this chapter, the salient features of the grid structure and the methods used in constructing the grid are described.

4.1 GRID STRUCTURE

Figure 1 shows a typical flow domain for a wing-body in physical space and its counterpart in computational space. The entire flowfield is transformed into a single rectangular block in the computational domain. The fuselage surface is transformed into a coordinate plane K=1, whereas the wing surface is unwrapped and transformed into a coordinate plane J=1. The natural lines on the configuration surface such as the wing-body intersection become lines of intersection of coordinate planes in the computational domain (fig. 2). The volume grid consists of C-meshes wrapped around the wing sections.

When the nacelle is added, it is convenient to divide the flowfield into two blocks (fig. 3). The structure of the volume grid in both blocks is essentially the same. The surface of the nacelle in physical space becomes a grid plane (K=constant) defining the boundary between the two blocks. The construction of the surface grid for the nacelle-wing intersection region is somewhat complicated when the wing-nacelle intersection ends abruptly in the middle of the wing. In that case it becomes very difficult to generate lines on the nacelle surface conforming to a smooth C-mesh. The following simple workaround effectively deals with this difficulty. A modified nacelle surface is defined in a manner illustrated in Figure 4. The hatched portion in Figure 4 is cut from the wing and attached to the nacelle and is defined to be a part of the nacelle surface. With this definition it is possible to define a smooth grid distribution. It is noted that this procedure is used to simplify the generation of the surface grid and does not constitute a compromise in the geometry.

4.2 GRID GENERATION TECHNIQUES

4.2.1 ALGEBRAIC GRID GENERATION

For simple, wing-alone geometries, the flow program calculates the grid internally using certain algebraic transformations. The details of the transformations may be found in Reference 8 and will not be repeated here.

4.2.2 NUMERICAL GRID GENERATION

The analytical grid generation techniques that work reasonably well for generating the volume grid over smooth wing-type surfaces can fail to produce smooth grids for cases where nacelles are present, in which case one must resort to numerical grid generation techniques[9,10]. Such techniques typically solve inhomogenous elliptic partial differential equations in the computational space with physical coordinates as dependent variables. The primary input to the three-dimensional grid generation program is the surface grid that serves as the boundary condition; a typical example of the surface grid is shown in Figure 5. The grid distribution within the flowfield is controlled via the inhomogenous source terms in the elliptic partial differential equations. The details of this method may be found in References 9 and 10.

5.0 TRANSONIC FLOW ANALYSIS

5.1 BASIC FORMULATION

5.1.1 GOVERNING EQUATIONS AND BOUNDARY CONDITIONS

The inviscid equations of mass, momentum, and energy written in conservation form

$$W_t + f_x + g_y + h_z = 0. \tag{1}$$

where the subscripts refer to the partial derivative with respect to that quantity and

$$W=\begin{Bmatrix}\varrho\\ \varrho u\\ \varrho v\\ \varrho w\\ \varrho E\end{Bmatrix} \quad f=\begin{Bmatrix}\varrho u\\ p+\varrho u^2\\ \varrho uv\\ \varrho uw\\ \varrho uH\end{Bmatrix} \quad g=\begin{Bmatrix}\varrho v\\ \varrho uv\\ p+\varrho v^2\\ \varrho vw\\ \varrho vH\end{Bmatrix} \quad h=\begin{Bmatrix}\varrho w\\ \varrho uw\\ \varrho vw\\ p+\varrho w^2\\ \varrho wH\end{Bmatrix} \tag{2}$$

and

$$E = p/(\gamma-1)\varrho + 1/2\, q^2, \qquad H = E + p/\varrho \tag{3}$$

are solved using a fourth order Runge-Kutta time stepping scheme by Jameson et al[7]. Here, the pressure p, the density ϱ, and the temperature T are normalized with respect to the freestream values.

The usual impermeable boundary condition is applied along all configuration surfaces such as fuselage, wing, and nacelle surfaces, i.e.,

$$\tilde{q} \cdot \hat{n} = 0. \tag{4}$$

together with a normal momentum relation to define the surface pressure p. Along the farfield boundary, characteristic boundary conditions utilizing Riemann invariants are employed to extract the necessary physical variables for proper flux formulation. More detailed discussions on the proper boundary condition formulation can be found in References 11, 12, and 13.

5.2 PROPFAN SIMULATION

The propeller is represented by an actuator disk along a computational grid plane. For a wing-mounted tractor, a natural choice for disk representation is a constant J plane located ahead of the nacelle, as shown in Figure 6a. For a wing-mounted pusher, constant I planes located downstream of the wing trailing edge are used to represent the actuator disk, as shown in Figure 6b. Within the present grid system, the propeller disk is represented by a number of rectangles along the grid plane, as illustrated in Figure 6. This number depends on the size of the grid as well as the grid distribution in the region where the propeller is to be located. For a typical grid containing 128 x 16 x 24 cells there are approximately 60 rectangles representing the disk.

Along the disk, appropriate boundary conditions are needed in order to simulate propeller power and swirl effects. Several ways of specifying disk boundary conditions have been explored. Two methods that have been proven effective are discussed in the following:

5.2.1 METHOD 1

It is assumed that the distribution of force components, F_x, F_y, and F_z, and the distribution of work, Q, done by the propeller are given. The velocity downstream of the disk, q_2, is extrapolated from the field. This velocity, q_2, and the momentum equations that contain the contributions of F_x, F_y, and F_z are used to solve for the velocity components u_2, v_2, and w_2, and the static pressure p_2 in terms of their upstream values of the velocity components u_1, v_1, and w_1, and pressure p_1, i.e.,

$$\varrho_2 U_2 u_2 + S_x p_2 = \varrho_1 U_1 u_1 + S_x p_1 + SF_x \tag{5a}$$

$$\varrho_2 U_2 v_2 + S_y p_2 = \varrho_1 U_1 v_1 + S_y p_1 + SF_y \tag{5b}$$

$$\varrho_2 U_2 w_2 + S_z p_2 = \varrho_1 U_1 w_1 + S_z p_1 + SF_z \tag{5c}$$

$$u_2^2 + v_2^2 + w_2^2 = q_2^2 \tag{5d}$$

$$\varrho_1 U_1 = \varrho_2 U_2 \tag{5e}$$

with S representing the disk surface area and S_x, S_y, and S_z the projection components of the cell face area in x, y, z directions, respectively. U is the flux velocity normal to the disk surface and is defined by

$$U = uS_x + vS_y + wS_z \tag{5f}$$

The energy equation containing the work term Q is used to solve for the density ϱ_2, i.e.,

$$\varrho_2 = 3.5\ p_2 / [H_1 + SQ/\varrho_2 U_2 - 1/2\ q_2^2\ ] \tag{5g}$$

The normal mass flux, ϱq_n, is calculated and used as a boundary condition for the upstream side of the disk. In essence, one specifies the distribution of force and work exerted by the propeller on the flow and the Euler solver calculates the corresponding flowfield.

$$M_2^2 = 5\ q_2^2 / (7T_t^2 - q_2^2) \tag{6a}$$

5.2.2 METHOD 2

It is assumed that the distribution of total pressure, p_t, total temperature, T_t, and the swirl, δ, are given at the downstream side of the disk. (These are sometimes available from an experimental measurement.) Downstream of the disk, the velocity, q_2, is obtained in the same way as in method 1. The Mach number, M_2, is computed from T_t and q_2 using the definition of total temperature

$$M_2^2 = 5\ q_2^2 / (7T_t^2 - q_2^2) \tag{6a}$$

The static pressure, p_2, is obtained from M_2 and p_t through the isentropic relation,

$$p_2 = p_t / (1 + 0.2\ M_2^2)^{3.5} \tag{6b}$$

Throughout this report, the specific heat ratio $\gamma = 1.4$ is used in all equations. With known values of δ and q_2, the velocity components u_2, v_2, and w_2 are easily computed. Finally, the density, ϱ_2, is obtained from the definition of the total enthalpy, H_2, and the normal mass flux, ϱq_n, is computed and used as a boundary condition for the upstream side of the disk. In essence, one applies flow conditions on the downstream side of the disk, and the Euler solver calculates the corresponding flowfield that is compatible with those conditions.

Since the solution process is iterative and not time accurate, one can encounter, temporarily, local regions of supersonic flow entering the disk, regions which usually disappear as the solution converges. These must be dealt with properly in order to assure stable convergence. For the case of supersonic upstream flow, i.e., when M_1 greater than 1, no boundary condition can be assigned along the upstream side of the disk (fig. 7). The flow variables needed for the calculations are obtained by extrapolation, and the normal mass flux is computed and used as a boundary condition downstream of the disk. For a subsonic downstream flow, i.e., when M_2 less than 1, this means one boundary condition described in both methods 1 and 2 must be relaxed in order to have a well-posed boundary value problem. In method 1, the work term, Q, is relaxed, while in method 2 the total temperature, T_t, is relaxed. In this way, the primary quantity of the propeller, the thrust, can still be properly simulated. In practice this situation can be encountered temporarily during the iteration process. Converged solutions usually exhibit subsonic flow entering the disk for cases which are representative of candidate installations. The present propeller disk theory is not valid for the case where both M_1 and M_2 are greater than 1.

Whitfield et al[13] have formulated a different approach for the propeller simulation. In their approach, the propeller inputs are treated as source terms in the form of body forces in the governing equations. We found that our implementation of that approach produced oscillatory solutions near the propeller region, especially for a heavily loaded propeller where the source terms were relatively large. In the present approach explained earlier, the propeller is treated as a boundary surface. The propeller inputs are implemented as boundary conditions so that discontinuous solutions across the disk are allowed and the propeller loading can be evaluated through the integration of momentum change across the disk.

5.3 EXHAUST FLOW SIMULATION

The exhaust plume can either be treated as a simplified solid body with the exhaust plume geometry obtained from an isolated exhaust flow analysis program[14], or solved simultaneously with the external flow through grid embedding. The former method is simple and is good for initial evaluations. However, it may not accurately account for the interaction between the exhaust flow and the external flowfield. The grid embedding approach, on the other hand, does account for the interaction between the two regions.

Various ways of obtaining detailed flowfield solutions around a specific domain can be found in the literature. These include local mesh refinement methods[15,16], zonal approaches[17,18], and a general grid overlapping method[19]. In the present study, a practical grid embedding method similar to the zonal approach that can be implemented into a three-dimensional flow analysis has been developed. Although the method is fairly general in nature, so far we have applied it only to the exhaust flow calculations. Certain general remarks regarding the algorithm issues are made after the specific application to the exhaust flow problem is described.

5.3.1 EMBEDDED EXHAUST FLOWFIELD

The basic concept is to define a common boundary between the embedded region and the global region. Then within each region, a separate well-distributed surface-fitted grid is constructed. Because a common boundary divides the regions, a finite volume formulation that conserves mass, momentum, and energy across the common boundary can be used without using sophisticated interpolation or extrapolation techniques. Boundary conditions similar to that of the method 2 in the propeller simulation, i.e., specifying the total pressure, total temperature, and the swirl, are used at the nacelle exit plane, and the flowfield inside the exhaust plume is solved using the same Euler technology as in the outer flow calculations. The treatment of the flux formulation and the implementation of the artificial dissipation terms along the common boundaries between the exhaust flow and the outer flow will be discussed in a later section. A similar concept has been used by Weatherill and Forsey[20] and Norton et al[21] in their study of complex aircraft and turbine cascade configurations.

To analyze the detailed exhaust flow and evaluate its effects on wing loading, we construct a grid inside the exhaust plume and solve the complete exhaust flowfield simultaneously with the external flow. A grid surface, positioned to lie in the vicinity of the exhaust plume boundary, forms a common boundary surface between the inner and the outer grids. A cylindrical grid is generated for the inner region using the same grid generation technology employed in the global flowfield. Typical grid for the exhaust flow is illustrated in Figure 8. Notice that the global flowfield and the exhaust flowfield share the same grid points along the exhaust plume surface.

5.3.2 GRID EMBEDDING

In this section we present certain ideas that could be pursued to make the grid embedding technology applicable to more general situations. The C-grid system used in the global flowfield analysis has an undesirable restriction on propeller disk representation; that is, the propeller is represented by a jagged polygon rather than by the circle illustrated in Figure 6. To remove such an undesired feature, a cylindrical grid system to produce an embedded grid at the nacelle-propeller region should be chosen as shown in Figure 10. An H-type grid may be used in the streamwise direction in order to represent the nacelle as well as the propeller better. Use of different grid systems in the global flowfield and the embedded flowfield introduces additional but manageable programming complexity in the flow solver. It needs a special formulation for the flux calculation in order to preserve mass, momentum, and energy fluxes at the common boundary. Two closely related issues to be considered in the code development process are grid topology and flow solver algorithm at the common boundary. These issues are discussed in the following:

5.3.3 GRID TOPOLOGY

For a multiple component configuration, it is likely that each component needs a specific grid system to facilitate the generation of surface-fitted grids. A simple way of generating grids for multiple component configurations is to produce grids around each component independently and let them overlap in the flowfield[19]. The major advantage of this approach is that grids can be generated easily. However, the drawback is that one has to develop an algorithm which can interpolate data back and forth between the grids about each component

in the overlapping region, a rather complex and formidable programming effort for a general three-dimensional application. An alternative method used in the present study is to define the grids on the common boundary surfaces between each region first, and then generate the field grid within each region. Because common boundaries between each region have been defined, communication of the flowfield between each region can be carried out through the common boundaries rather than the overlapping regions, a significant simplification of data management between each embedded flowfield region. For example, surface interpolations reduce to line interpolations for two-dimensional flow problems, and volume interpolations reduce to surface interpolations for three-dimensional flow problems in this common boundary grid embedding approach.

5.3.4 FLOW SOLVER ALGORITHM AT COMMON BOUNDARY SURFACES

The most important consideration in the flow solver algorithm is to achieve precise flux conservation across the common boundary. Shock waves and contact discontinuities can only be correctly captured across the common boundary if the physical fluxes, i.e., mass, momentum, and energy fluxes, are exactly balanced across the boundary. In the case of exhaust flow grid embedding explored in the present study, identical grid points are used along the common boundary between the exhaust flow and the global flowfield, as shown in Figure 10. The simplest way of calculating fluxes at a point P along the boundary is to take the average fluxes at the interior point A and the exterior point B. These flux quantities are used both in the exhaust flow and the global flowfields to ensure an exact flux balance between two regions. Since rapid changes in grid distributions between the two regions can occur, the simple averaging procedure used in the exhaust flow calculations, although satisfying the conservation law, may suffer a certain degree of numerical inaccuracy.

Another way of calculating flux at the common boundary surface would be to use a nodal point scheme, as shown in Figure 11, where the physical variables are defined at the grid point rather than at the cell center. Within each grid cell ABCD, a bilinear or trilinear formula may be used to define the physical variables. The flow variables at any point within that cell including the boundary can be evaluated within the accuracy of the basic formulation. A secondary cell which surrounds the grid point can be constructed for the formulation of flux conservations. We think this nodal point finite element-type formulation might have a possible advantage of reducing grid sensitivity on the flow solution over the cell centered simple averaging scheme. However, we have not implemented this to date, since our present cell-centered scheme works quite well.

Another factor that affects the accuracy of the numerical solution is the treatment of the second order and fourth order dissipation terms at the common boundary. A method that has been proven reliable is to set all dissipation terms to zero at the common boundary and gradually scale back to the original level away from the boundary both in the embedded region and in the global region.

5.4 BOUNDARY LAYER COUPLING

In the following, we present certain salient features of the approach used in the present program. More details may be found in Reference 22.

5.4.1 INVISCID MODELING

The effect of the boundary layer may be simulated in two ways. In one, the boundary layer displacement thickness, $\delta,^*$ is added to the configuration surface and the revised configuration is analyzed with the boundary condition given by Equation 4. This approach requires construction of a new grid over the revised configuration after every boundary layer calculation. In the other, the boundary layer effect is simulated by an approach suggested by Lighthill[23]. The surface boundary condition is replaced by a surface transpiration condition given by:

$$\sigma_{2D} = \varrho u_n = d/ds\ (\varrho u_s\ \delta^*)\ \text{in 2D} \tag{7}$$

where σ denotes the transpiration source term, u_n is the normal velocity, u_s is the tangential velocity in 2D. This so-called transpiration approach is generally preferred since the grid about the configuration need only be calculated once. Both approaches are equivalent when the surfaces over which boundary layers are calculated have small curvature.

When the outer inviscid flow is modeled by full-potential equations, specification of transpiration mass sources is sufficient to provide a unique solution. In the solution of the Euler equations, however, it is also necessary to specify the momentum and energy of the transpiration flow. These may be derived by using an approach suggested by Johnston and Sockol[24]. In 2D these are given by:

$$(\varrho u_n\ u_s)_o = (u_s)_o\ \sigma = (u_s)_o\ d/ds\ (\varrho u_s \delta^*) \tag{8}$$

$$(\varrho u_n u_n)_o = (u_n)_o\ \sigma = (u_n)_o\ d/ds\ (\varrho u_s\ \delta^*) \tag{9}$$

and

$$(\varrho u_n H)_o = (H)_o\ \sigma = (H)_o\ d/ds\ (\varrho u_s \delta^*) \tag{10}$$

These equations were also given by Whitfield[13]. They are written in a boundary conforming coordinate system to give

$$\text{x momentum source} = (u)_o \sigma \tag{11}$$

$$\text{y momentum source} = (v)_o \sigma$$

where, σ is given by Equation 7. These equations suggest that the mass source is a common factor among all transpiration quantities. The momentum and energy transpiration may be derived by multiplying the mass transpiration by an appropriate component of the velocity or by the enthalpy. The pressure contribution to the momentum balance of the cells next to the surface is determined by extrapolation.

Even though the approach by Johnston and Sockol provides the expressions for the momentum and energy source terms, the values to be used at the surface remain to be established. Murman[25] has discussed various alternative locations at which viscous/inviscid matching boundary conditions may be applied. He suggests that when matching boundary conditions are specified at the wall, the choice of all but one condition is arbitrary since the surface generally happens to be an inflow boundary in a coupled problem. Equation 7 may be

be considered to be the nonarbitrary condition to be specified at the wall. The other conditions must be chosen on the basis of physical reasoning. Any values chosen will produce a unique Euler solution. The issue is to choose values that will create a flow that is representative of the one that is being modeled. Our choice is to choose local values of the tangential velocity and enthalpy which are extrapolated from the outer flow field.

5.4.2 INTEGRAL BOUNDARY LAYER METHOD IN 2D

The present method includes a 2D strip theory boundary layer, on the wing, coupled with the outer flow. In 2D, the boundary layer is calculated by Green's lag-entrainment method[26]. It is an integral method that calculates the boundary layers and wakes. The turbulence modeling is derived from the Bradshaw-Ferris turbulent energy equation[27]. The method had proven to be fast and reliable for computing turbulent boundary layers about airfoils.

Like all integral boundary layer methods, Green's method can be written in the form

$$A\, d\delta^*/ds = B\, du_i/ds + C \tag{12}$$

If A = 0, it is impossible to use Equation 12 to solve for $d\delta^*/ds$ in terms of du_i/ds. In this case it is not appropriate to integrate the boundary layer equations using a prescribed value of du_i/ds. This is referred to as the separation singularity. The separation singularity can be avoided by integrating the boundary layer equations with a prescribed value of $d\delta^*/ds$ and solving for du_i/ds. This is the so-called inverse mode of calculating the boundary layer flow. The inverse mode allows one to use the boundary layer equations even in regions of mildly separated flows.

There exist several different inverse modes. The general idea is to prescribe some quantity other than the velocity to the boundary layer and allow the boundary layer equations to generate a viscous version of the velocity, u_v. In these calculations, $m = \rho u_i \delta^*$ is prescribed to the boundary layer since it was shown by LeBalleur[28] that the boundary layer calculations will then be free of singularities.

Although the inverse boundary layer method explained above allows one to calculate boundary layers in a stable manner for a wide range of interesting flow conditions, there remains the problem of coupling the viscous solution with the inviscid solution. That is, how to adjust the prescribed value of m so that the viscous velocity calculated at the outer edge of the boundary layer solution agrees with the inviscid velocity calculated by the outer inviscid solution.

In the present calculations, the following update scheme proposed by Carter is used:

$$m^{n+1}/m^n = \omega u_v/u_i + (1-\omega) \tag{13}$$

where ω is a relaxation factor. The subscripts v and i refer to the viscous and the inviscid versions of the variable and the superscript n refers to the iteration level. This simple update scheme has proven to be remarkably effective.

6.0 PROGRAM DEVELOPMENT AND VECTORIZATION

In this chapter we describe some details regarding the computer program development and the vectorization of the program to run on the Cyber 205 computer at NASA-Langley.

6.1 GRID GENERATION PROGRAMS

The grid generation programs used in this method were originally developed by Yu[9,10] at Boeing. The basic program development was done on CYBER 175 and CRAY-1S machines. The use of these programs for various problems of interest is well documented[6,9,10]. The major changes to the grid generation code made during the contract relate to the NASA-Langley wing-nacelle test case. The original code could generate the grid for conventional aircraft configurations, whereas the present code can also generate grids for a model with endplates or model with sidewalls. The grid generation programs are expected to be executed in scalar mode on CYBER 205 at NASA-Langley. It was decided not to vectorize these codes for the CYBER 205 since the total resources spent in generating grids in a given problem are quite small.

6.2 FLOW PROGRAMS

The flow program used in this method originates from the FLO57 code developed by Jameson. Several modifications were made to the original version of this program to make it more robust and reliable and to improve accuracy of the solution. These changes are well documented[11,29]. The major changes made to this program during this contract are now described.

6.2.1 ALGORITHM CHANGES

First, the program was modified to handle slipper nacelles mounted on wings. Such a nacelle is primarily represented by a double K plane at the common boundary of the two blocks (see Chapter 4). The program was changed to handle the no-flow boundary condition over a portion of this grid plane.

The second major change made during this contract relates to implementing the propeller simulation. The proper propeller boundary conditions to simulate the propeller were incorporated in the program. Both pusher and tractor propeller configurations can be modeled.

All the code changes were made on the Boeing CRAY, primarily to facilitate turnaround and expedite the modification process. This was followed by conversion to the Cyber 205 and optimization for that computer.

6.2.2 CYBER VECTORIZATION

Once the algorithm modifications stabilized, the code was then transported to Cyber 203 at NASA-Langley where the initial part of the vectorization of the code was done. When it was learned that the Cyber 203 was going to be replaced by Cyber 205, that part of the work was stopped until the Cyber 205 machine was available. In the interim, Control Data Corporation (CDC), which was to supply the Cyber 205 to NASA-Langley, made its own Cyber 205, located at Minneapolis, Minnesota, available for continuation of the vectorization work.

The following code modifications were made to the program to vectorize the code on the CDC Cyber 205:

1. The CYBER 203/205 vectorization requires that the vectors which are being operated on be contiguous in the memory. In order to conform with this requirement, the major dimensioned variables in the code had to be redimensioned. For example, W(6,IDX,IDY,IDZ) had to be redefined as R(IDX,IDY,IDZ), RU(IDX,IDY,IDZ), etc. This was a significant change to the code since these variables are used extensively in the program.

2. In order to take advantage of the longer vector lengths, it was decided to follow an approach taken by M. Hodus of Control Data Corporation. In this approach, an entire 2D grid plane is operated upon at a time. Certain bit vector patterns are defined to mask the boundary data where all the calculations are not required. Typical masking pattern for a 2D plane is shown in Figure 12a. The 2D vectorization also required a major modification of the code due to the extensive use of the vector syntax. The three major subroutines which took most of the CPU time were essentially rewritten. Typical effects of vectorization on the performance of the code is illustrated in Figure 12b. Here, the dramatic reduction in the percentage of CPU time taken by the EULER subroutine before and after vectorization is evident. After vectorization, the percentage of CPU time due to Filter becomes more important because it is now a greater percentage of the total. The availability of the Hodus model made the process fairly straightforward. Also, the availability of the CDC machine at Minneapolis which provided excellent turnaround helped the vectorization process further.

The performance of the code for a typical case is summarized in Table 1. It is apparent that the vectorized version on CYBER 205 is about 40 % faster than the Boeing CRAY 1S version of the code for the grid size of 128*16*24. However, vectorization makes the mesh refinement more difficult since the 3D arrays defining various variables in the program are required to be redefined for every mesh. The mesh refinement, which is a way of accelerating the convergence of the solution, may be performed by executing the code twice. The coarse mesh calculations are performed using a program binary which is dimensioned to its length. The fine mesh calculations are then performed using another program binary which is dimensioned for fine mesh. The second execution can take the flowfield calculated in the first run and refine it before starting the fine mesh iterations. This method is somewhat cumbersome, but avoids having to rewrite the program completely in order to redefine the dimensioned variables completely and pass them as arguments in subroutine calls.

7.0 RESULTS OF TEST CASES

Several test cases have been analyzed using the Euler code described here to demonstrate its capability to solve for the aerodynamic characteristics of propfan-airframe installations. General information for these cases is given briefly in Table 2. The embedded grid approach was used in the NASA test model case. The results from these solutions are described next.

7.1 CASE 1: NASA-AMES TEST MODEL

The NASA-Ames test model[4] was analyzed with and without propeller effects and the results from the solution were compared to test data. Both propeller-off and propeller-on analyses were performed for this wing-mounted tractor configuration. A simplified solid plume model was used in the initial code development. A total of 128 x 16 x 24 cells was used in the analysis, with 80 x 16 cells on the wing, 104 x 10 cells on the fuselage, and 128 x 6 cells on the nacelle and the exhaust plume. In the exhaust plume grid embedding study, an additional 30 x 6 x 12 cells were used in the exhaust flow region, where the global grid cells were reduced to 112 x 16 x 24 in order to fit the complete program into the 2 MW Boeing CRAY installation.

To simulate the propeller power and swirl effects, the total pressure and the swirl angle distributions downstream of the propeller obtained from experiment, shown in Figure 13, were used as input for the disk boundary conditions[29]. The swirl angle is considered to be positive if the propeller rotates upward inboard of the nacelle. The total temperature needed for the disk simulation was calculated from the total pressure distribution using the isentropic relation. Figures 14a and 14b compare the wing surface pressure distributions of the propeller-off calculations with test data. The presence of the nacelle, which causes a high pressure peak inboard of the nacelle, has been well-simulated in the present study. Figures 15a and 15b show the results of the propeller-on calculations with the propeller rotating up-inboard of the nacelle. The swirl effects causing an increase of angle of attack inboard of the nacelle and resulting in an increase of shock strength have been correctly predicted in the present analysis. Both the propeller-off and the propeller-on results show slight oscillations in wing surface pressure distributions. The major cause of such oscillations is nonsmoothness of the wing geometry. Similar oscillations can also be observed in the test results. Figures 16a and 16b show the results of the same configuration with the propeller mounted downstream of the wing trailing edge to simulate a wing-mounted pusher arrangement. The wing surface pressure distributions are much less affected by the propeller of this pusher configuration than that of the tractor configuration. However, the benefit of swirl recovery due to the presence of the wing for the tractor configuration diminishes for this pusher configuration, as can be seen clearly from the cross flow velocity vector plot in the wing wake region shown in Figures 17a and 17b. Such swirl recovery may contribute a significant amount of efficiency improvement, and is an interesting subject which needs further investigation. Another approach for swirl recovery is the use of counterrotating propellers.

Due to the complexity of the turboprop configuration, our initial attempt in grid embedding research was limited to the evaluation of the exhaust effects on the wing loading characteristics. Figures 18a, 18b, and 18c show the velocity vector plot and the Mach number contour plot in the exhaust flow region for a high total pressure and total temperature exhaust condition. It can be seen clearly that a supersonic exhaust jet can be simulated with the present grid embedding approach. The exhaust conditions were set to high values to test the robustness of the program and do not necessarily correspond to a real exhaust condition. Figures 19a and 19b compare the results of the present exhaust condition with the results of a solid plume calculation. The expansion of the exhaust flowfield downstream of the nacelle exit plane causes the flowfield near the trailing edge to accelerate both inboard and outboard of the nacelle, an expected trend from the inviscid theory. No extensive evaluation has been made yet to check the detailed exhaust plume effects on wing loadings.

7.2 CASE 2: ONERA M6 WING

The next test case involved the analysis of the Onera M6 wing at M=.84 and α=3.06 deg. The analyses were performed with and without the boundary layer coupling. Typical section C_p results are presented in Figure 20. The effect of the boundary layer on the wing surface distribution is apparent from these figures.

7.3 CASE 3: NASA-LANGLEY WING-NACELLE MODEL

The third test case involved the analysis of the model tested at NASA-Langley[30]. This model was held between endplates. The typical surface grid is shown Figure 21. The flow analysis results with and without the propeller are shown in Figure 22. The propeller loading used in these calculations came out of Reference 31 and is shown in Figure 23. Figure 23a shows the wing alone results with and without boundary layer at M = .75 and α = 3.0 deg. The overall suction level has been predicted properly. However, the shock location is too far aft. Apparently the stripwise boundary layer approach underpredicts the effect on shock location and the lower surface pressure distribution. Figures 23b and 23c show the nacelle on and propeller off and on results respectively at a location in the propeller wash region. The agreement between test and theory appears to be good except near the shock.

8.0 CONCLUSIONS AND RECOMMENDATIONS

Based on the results presented in Chapter 7, it is concluded that the Euler-based method of solving the flow about a propfan configuration has been developed and is ready for use. The code can handle various types of geometries including the wing alone, wing-body, and wing-body-nacelle combinations. Propfan effects are treated as an actuator disk. Boundary layer effects are accounted for using a stripwise boundary layer coupling approach.

It is recommended that this code be exercised further on more cases at different prop loading conditions and with different grid densities to validate its applicability further.

9.0 REFERENCES

1. Boeing Commercial Airplane Co., "Energy Consumption Characteristics of Transports Using the Prop-Fan Concept," NASA Contract Report 137937, NASA, Moffett Field, CA, October 1976.
2. Welge, Robert H.; and Crowder, James P., "Simulated Propeller Slipstream Effects on a Supercritical Wing," NASA CR-152138, Douglas Aircraft Co., Long Beach, CA, June 1978.
3. Welge, H. R.; Neuhart, Dan H.; and Dahlin, John A., "Analysis of Mach 0.8 Turboprop Slipstream Wing/Nacelle Interactions," NASA CR-166214, Douglas Aircraft Co., Long Beach, CA, July 8, 1981.
4. Smith, Ronald C.; and Levin, Alan D., "Propfan Installation Aerodynamics of a Supercritical Swept Wing Transport Configuration," AIAA paper 81-1563 presented at AIAA/SAE/ASME 17th Joint Propulsion Conference, July 27-29, 1981, Colorado Springs, CO.
5. Levin, Alan D.; and Smith, Ronald C., "Propfan Propulsion Integration Test Techniques," AIAA Paper 82-0577 presented at AIAA 12th Aerodynamic Testing Conference, March 22-24, 1982, Williamsburg, VA.
6. Samant, Satish S.; Hultman, Don; Lines, Ray; Yu, N. J.; and Rubbert, Paul E., "Prop-Fan Nacelle Installations at Transonic Speeds," NASA CR-166376, October 1982.
7. Jameson, A.; Schmidt, W.; and Turkel, E., "Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes," AIAA-81-1259, 1981.
8. Thompson, J. F.; Thames, F. C.; and Mastin, C. M., "Automatic Numerical Generation of Body-Fitted Curvilinear Coordinate System for Field Containing any Number of Arbitrary Two-Dimensional Bodies," J. Computational Physics, vol. 15, pp. 299-319, 1974.
9. Yu, N. J., "Grid Generation and Transonic Flow Calculations for Three-Dimensional Configurations," AIAA Paper 80-1391, July 1980.
10. Yu, N. J., "Transonic Flow Simulations for Complex Configurations with Surface Fitted Grids," AIAA Paper AIAA-81-1258. Presented at the AIAA 14th Fluid and Plasma Dynamics Conference, June 23-25, 1981, Palo, Alto, CA.
11. Chen, H. C.; Yu, N. J.; Rubbert, P. E.; and Jameson, A., "Flow Simulations for General Nacelle Configurations Using Euler Equations," AIAA-83-0539, 1983.
12. Jameson, A.; and Baker, T. J., "Solution of the Euler Equations for Complex Configurations," AIAA-83-1929, 1983.
13. Whitfield, D. L.; and Jameson, A., "Three-Dimensional Euler Equation Simulation of Propeller-Wing Interaction in Transonic Flow," AIAA-83-0236, 1983.
14. Peery, K. M.; and Forester, C. K., "Numerical Simulation of Multistream Nozzle Flows," J. AIAA, vol. 18, 1980.
15. Usab, Jr., W. J.; and Murman, E. M., "Embedded Mesh Solutions of the Euler Equations Using a Multiple Grid Method," AIAA-83-1946, 1983.
16. Berger, M. J.; and Jameson, A., "Automatic Adaptive Grid Refinement for the Euler Equations," MAE Report No. 1633, Princeton University, 1983.
17. Rai, M. M., "A Conservative Treatment of Zonal Boundaries for Euler Equation Calculations," AIAA-84-0164, 1984.

18. Cambier, L.; Ghazzi, W.; Veuillot, J. P.;and Viviand, H., "A Multidomain Approach for the Computation of Viscous Transonic Flows by Unsteady Type Methods," Editor: W. G. Habashi, Recent Advances in Numerical Methods in Fluids, vol. 3, Pineridge Press, Swansea, U.K., 1983.
19. Benek, J. A.; Steger, J. L.;and Dougherty, F. C., "A Flexible Grid Embedding Technique With Application to the Euler Equations," AIAA-83-1944, 1983.
20. Weatherill, N. P.;and Forsey, C. R., "Grid Generation and Flow Calculations for Complex Aircraft Geometries Using a Multiblock Scheme," AIAA-84-1665, 1984.
21. Norton, R. J. G.; Thompkins, Jr., W. T.; and Haimes, R., "Implicit Finite Difference Schemes with Nonsimply Connected Grids A Novel Approach," AIAA-84-0003, 1984.
22. Samant, S. S.;and Wigton, L. B., "Coupled Euler/Boundary Layer Analysis in Transonic Flow," AIAA Paper 83-1806, AIAA Applied Aerodynamics Conference, July 13-15, 1983, Danverse, MA.
23. Lighthill, M. J., "On Displacement Thickness," Journal Fluid Mechanics, vol. 4, pt. 4, pp. 383-392, Aug. 1958.
24. Johnston, W.; and Sockol, P., "Matching Procedure for Viscous-Inviscid Interactive Calculations," AIAA Journal, vol. 17, no. 6, pp. 661-663, June 1979.
25. Murman, Earll M.; Bussing, Thomas R. A., "On the Coupling of Boundary Layer and Euler Equation Solutions," 2nd Symposium on Numerical and Physical Aspects of Aerodynamic Flows, Jan. 17-20, 1983, Long Beach, CA.
26. Green, J. E.; Weeks, D. J.;and Brooman, J. W. F., "Prediction of Turbulent Boundary Layers and Wakes in Compressible Flow by a Lag-Entrainment Method," R. and M. no. 3791, British A.R.C., 1977.
27. Bradshaw, P.; and Ferris, D., "Calculation of Boundary Layer Development Using Turbulent Energy Equations in Compressible Flow on Adiabatic Walls," J. Fluid Mechanics, vol. 46, 1977.
28. LeBalleur, J. C., "Couplage Visquex-Non Visquex: Methods Numerique et Applications Aux Ecoulements Bidimensionals Transonniques et Supesnniques," La Recherde Aerospatialle, Mar-Apr 1978.
29. Yu, N. J.; Samant, S.S.; and Rubbert, P.E., "Flow Prediction for Prop-Fan Configurations Using Euler Equations," AIAA Paper 84-1645, AIAA 17th Fluid Dynamics, Plasma Dynamics, and Laser Conference, June 1984, Snowmass, CO.
30. Rohrback, Carl, "A Report on the Aerodynamic Design and Wind Tunnel Test of a Prop-Fan Model," AIAA Paper 76-667, July 1976.
31. Bartlett, G. R., "An Experimental Investigation of Propfan Installation on an Unswept Wing," M.S. Thesis, George Washington University, Washington D.C., August 1983.

ACKNOWLEDGMENTS

The authors wish to thank Ted Reyhner of The Boeing Company for his assistance in making them familiar with the CYBER 203 and 205 systems during the initial period of the contract. The authors greatly appreciate the help received from John Gee of Control Data Corporation for his valuable assistance in using the CDC system at Minneapolis, Minn. The authors also wish to thank Control Data Corporation for allowing us to use their system to vectorize the code in an efficient manner.

A.0 SYSTEM EXECUTION

The transonic analysis method described earlier constitutes a system of computer programs that must be executed properly to arrive at meaningful final results. The two major segments of the system are (1) grid generation, and (2) transonic flow analysis, with or without boundary layer.

Table A1 provides a summary of the typical steps involved and Figure A1 provides a flowchart. The job control decks to be used in all the execution steps are shown in Appendix C. The steps involved in these segments are described below:

A.1 GRID GENERATION

The primary purpose of this segment is to construct a smooth well-distributed grid over which the flow equations are to be solved. The major steps involved in this segment are listed below:

A.1.1 STEP 1: PROCESSING THE GEOMETRY INPUT FILES

In this step, input files prepared by the user are read in to produce certain intermediate files that are to be input to the volume grid generation program. The user must prepare one or two such files depending on whether he is trying to analyze a wing-body or a wing-body-nacelle configuration. Typical preprocessing steps are described below.

a. A preprocessor program (PRGRD11) is used to read in a user-prepared file (INPUT1A) to generate data files containing the geometry information for the fuselage and the wing. If the nacelle is present, this step generates the data files for the geometry of the fuselage, the inboard wing, and the nacelle.

b. This step is performed only if the nacelle is present in the configuration. A preprocessor program (PRGRD12) is used to read in the user-prepared input file (INPUT1B) to generate a data file containing the geometry information for the outboard portion of the wing.

c. This step is required only if the nacelle is present. A preprocessor program (PRGRD13) is used to combine the wing geometry files prepared in steps 1a and 1b above into one single file.

A.1.2 STEP 2: VOLUME GRID GENERATION

This step involves execution of the 3D grid generation program which uses the files generated in the Step 1. Typically two or three grid meshes are generated during this step. The flow program normally uses the finest of these grids. This step requires an input file INPUT2.

A.1.3 STEP 3: VISUAL EXAMINATION OF THE GRID

The success of the flow analysis depends to a large extent on the smoothness of the grid over which the equations are solved. The importance of the smoothness of the grid cannot be overemphasized. Hence, it is highly recommended that the user examine the grid visually prior to proceeding with the flow analysis. At this time, the only effective way in which the smoothness of the grid can be ensured is via visual inspection. A separate postprocessing program (PRPOST) that reads in the grid and a small input file (INPUT3) may be used to extract the grid planes in network form, which may be displayed graphically using any standard programs that can display arbitrary 3D network geometries.

A.2 TRANSONIC FLOW ANALYSIS

Once the volume grid is constructed to user's satisfaction, the next major segment is the execution of the transonic flow program. The major steps involved in this segment are described below:

A.2.1 STEP 4: EXECUTION OF THE FLOW PROGRAM

Execution of the program (PRFLOW) may be performed in two different modes.

a. Startup mode: In this mode the user starts the flow solution in the coarsest mesh and continues to a desired degree of convergence. The solution is continued in finer meshes to desired convergence by starting with the interpolated solution in the coarser mesh. When the finest available mesh is reached, the user may use that solution to extract the aerodynamic properties of the flowfield. It is highly recommended that the user store the flowfield calculated in this step. This flowfield may then be used in future to restart the solution. This step requires an input file INPUT4A.

b. Restart mode: In this mode the user starts with a previously stored solution and continues the solution in the fine mesh to higher degree of convergence. Again it is recommended that the user store the flowfield calculated at the end of this step. This step requires an input file INPUT4B.

A.3 FLOWFIELD POST PROCESSING

A.3.1 STEP 5: FLOWFIELD INTERROGATION

Once the flowfield calculations have been completed, the user may interrogate the solution by using the postprocessing program (PRPOST), which requires an input file (INPUT3).

B.0 PREPARATION OF INPUT FILES

Perhaps the most tedious part of executing these programs is the preparation of the input data required by the various program elements. The following pages describe the files that the user is required to prepare. These files are identified as INPUTnx where nx is the step in which it is used. The input format associated with each file is also provided.

B.1 PREPARATION OF FILE INPUT1n

This file is required as an input to the grid generation preprocessor program PRGRD11 in Step 1a and contains the geometric curves describing the fuselage and the airfoil sections for the wing. If the nacelle is present, then the file contains only the part of the wing between the fuselage and the nacelle, and the inside and outside nacelle. In addition, certain other control parameters are required by the program for proper execution. These inputs are briefly described below.

B.1.1 GENERAL INPUT

This part of input includes such information as the desired size of the grid (number of cells to be constructed in each direction of the computational grid), the number of meshes, controlling parameters that tell the program which components are present and whether the component intersections are to be input, etc.

B.1.2 FUSELAGE SECTION CURVES

These are basically constant x cuts on the fuselage. The input description for these curves includes the x coordinate where these cuts are located, and for each section a set of the (z,y) coordinate pairs.

B.1.3 WING SECTION CURVES

The wing section curves are essentially airfoil sections located at various spanwise stations. Each airfoil section is expected to be nondimensionalized by local chord length. The coordinates of the leading edge point of the section (which happens to be the origin of the airfoil section in the nondimensionalized coordinate system) referred to in the global system, the physical chord of the section, and certain scaling factors are also included as input. Certain wing sections such as the wing-body intersection and the wing-nacelle intersection may require the spanwise variation to complete the accurate definition of that section.

B.1.4 NACELLE CURVES

Nacelle sections are very similar to fuselage sections. Section cuts on each side of the nacelle are input separately. The nacelle sections are required to lie in constant x planes in the global reference system (i.e. they must incorporate any pitch and toe-in that exists in the physical nacelle). If the nacelle is modified in a manner described in Chapter 4, then input in this file must be sections of the modified nacelle.

B.1.5 INPUT FORMAT FOR FILES INPUT1n

The input format of the file INPUT1n is now described. The cards Fn relate to the geometric curves describing the fuselage, the cards Wn relate to the wing, and the cards Nn relate to the inside and outside nacelle. Cards Gn are required in the beginning of the file to provide the program with the control parameters. All the header cards in this file are read in by a format (1X) and all the data cards are read by a format (8f10.0).

FILE INPUT1A

Card No.	Column No.	Name	Description
			GENERAL DATA
G1	1-80	TITLE	An 80-column title.
G2	1-80		Header card. Header cards are essentially dummy cards provided for identification of the data in the following cards. Typically the fields in the header cards should include the generic names of the variables to be included in that field in the following card(s). The header cards are read in by format (1x), which means that they may contain any legal characters including blanks. A typical header card is shown below.

FNX FNY FNZ FMESH FSCR

The header card is provided for easy identification of the input variables in the input file without having to look into the Fortran code in the program. FNX, FNY, etc. In the above line are the names of the variables to be read on the following card. It is not imperative that the header card be typed exactly since it is not read in by any rigid format in the program.

G3	1-10	FNX	Number of grid cells in the I direction in the computational domain in the coarsest mesh. In physical domain this number is equivalent to the number of cells in the chordwise direction of the wing and includes the cells on the entire surface of the wing as well as the wake.
	11-20	FNY	Number of grid cells in the J direction in the computational domain in the coarsest mesh. In the physical plane this corresponds to the number of cells in the direction normal to both the chord and the span.
	21-30	FNZ	Number of grid cells in the K direction in the computational domain in the coarsest mesh. In the physical plane this corresponds to the number of cells in the spanwise direction.
	31-40	FMESH	Number of meshes. In Euler analyses only the fine mesh grid is used. Hence the actual size of the mesh will be $FNX = FNX^{(FMESH-1)}$ etc.

Card No.	Column No.	Name	Description
	41-50	FSCR	I/O unit on which some of the intermediate results are written. If set to 0. then the program puts that information on TAPE13.
G4	1-80		Header card.
G5	1-10	FFUS	Parameter specifying the presence of the body geometry data in the input file. =0 if the body is absent. =1 if the body is present.
	11-20	FNAC	Parameter specifying the presence of the nacelle geometry data in the input file. =0 if the nacelle is absent. =1 if the nacelle is present.
	21-30	FWBIN	Parameter specifying the presence of the wing-body intersection geometry in the input file. =0 if the wing-body intersection is assumed to be flat. =1 if the wing-body intersection is to be input with the first wing section.
	31-40	FWNIN	Parameter specifying the presence of the nacelle-wing intersection geometry in the input file. =0 if the wing-nacelle intersection is assumed to be flat. =1 if the wing-nacelle intersection is to be input with the last wing section.
G6	1-80		Header card.

Card No.	Column No.	Name	Description
G7	1-10	FWDIS	Parameter specifying the spanwise distribution of the wing surface grid. =0 if the spanwise wing surface grid stations are to be placed at equal intervals. =1 if the spanwise wing surface grid stations are to be calculated using cosine rule. =2 if the spanwise stations in the wing surface grid are to be input by the user. User must input cards G8-G11 if FWDIS is 2.
	11-20	FSB	Factor used in the calculation of the cosine spacing. Used only if FWDIS=1.
	21-30	FST	Factor used in the calculation of the cosine spacing. Used only if FWDIS=1. NOTE: Input cards G8-G11 only if FWDIS=2.
G8	1-80		Header card.
G9	1-10	NSPAN	Number of spanwise grids. The user must input card G11 NSPAN times.
G10	1-80		Header card.
G11	1-10	ZSPAN	Spanwise location of the wing surface grid. This card must be input NSPAN times.
			FUSELAGE DATA
F1	1-80		Header.
F2	1-10	FIFUS	Number of input curves to describe the fuselage geometry. NOTE: Each fuselage section curve consists of a number of points in a constant x plane. The set of cards F3-F6 must be repeated FIFUS times. The curves input to this program may be double valued in y coordinate but cannot double valued in z coordinate (i.e., these curves are expected to be monotonic in z coordinate).

Card No.	Column No.	Name	Description
F3	1-80		Header.
F4	1-10	XF	x coordinate at which the curve is located.
	11-20	FN	Number of points on the curve.
	21-30	SAME	Parameter indicating whether the previous curve is to be reused at the present body station. NOTE: This facilitates input of several constant sections without having to provide the section curve cards (F5-F6) repeatedly. =0 if the cards F5-F6 describing a new section are input. In this case the card F6 is repeated FN times. =1 if the previous section is to be used. NOTE: Input cards F5-F6 only if SAME=0.
F5	1-80		Header. NOTE: Card F6 must be repeated FN times if SAME is 0.
F6	1-10	YP	z coordinate of a point on the curve.
		ZP	y coordinate of a point on the curve.

WING DATA

W1	1-80		Header.
W2	1-10	FNS	Number of wing sections to be input. Maximum 21.
	11-20	SWEEP	Leading edge sweep angle.

Card No.	Column No.	Name	Description
	21-30	DIHED	Dihedral angle.
			NOTE: Each wing section is input in its nondimensionalized form through the x/c and y/c coordinates referred to a coordinate system attached to the leading edge of the section. The upper and the lower surface of the section are input separately. In order to locate this section in the global coordinate system, the coordinate of the origin point (the leading edge) is also given in the global coordinate system along with certain scaling and pitching parameters. There must be FNS sets of cards W3-W10.
W3	1-80		Header.
W4	1-10	ZLE	z coordinate of the leading edge of the section to be input.
	11-20	XL	x coordinate of the leading edge of the section to be input.
	21-30	YL	y coordinate of the leading edge of the section to be input.
	31-40	CHORD	Chord length of the section being input.
	41-50	THICK	Thickness scaling factor for the section. The input airfoil y/c coordinate is multiplied by this factor to calculate the airfoil that is actually used in generating the surface grid. We do not recommend the use of this parameter. Set value to 1.
	51-60	AL	Pitching angle of the section. The airfoil is always pitched about the leading edge of the section. It is recommended that the user not use this feature of the input unless he is exactly sure of the effect of the rotation introduced because of this input.
	61-70	FSEC	Parameter to indicate whether the previous section is to be repeated.
			=0 new section on cards W5-W8 is expected.
			=1 previous section is to be used.
			NOTE: Cards W5-W10 are to be provided only if FSEC=0.

Card No.	Column No.	Name	Description
W5	1-80		Header.
W6	1-10	YSYM	Parameter indicating the symmetry of the section. =0 if the section is not symmetric. Both the upper and lower section curves are to be input. =1 if the section is symmetric. Only the upper section curve may be input. The lower section curve is derived by inverting the upper section curve about the origin and the x/c axis.
	11-20	FNU	Number of points on the upper section curve.
	21-30	FNL	Number of points on the lower section curve. NOTE:The FNL may be different from FNU only if YSYM is 0. If YSYM is 1., then the FNU and FNL must be equal.
W7	1-80		Header. NOTE: Card W8 must be repeated FNU times.
W8	1-10	XPU	x/c coordinate of the point on the upper section curve. The section must always start with the leading edge as the origin.
	11-20	YPU	y/c coordinate of the point.
	21-30	ZPU	Spanwise coordinate of the point in case the section represents a nonplanar intersection between the wing and the fuselage or the wing and the nacelle. NOTE: Cards W9-W10 are to be input only if YSYM=0.
W9	1-80		Header. NOTE: Card W10 must be repeated FNL times.

Card No.	Column No.	Name	Description
W10	1-10	XPL	x/c coordinate of the point on the lower section curve. The section must always start with the leading edge as the origin in the x/c and y/c coordinates.
	11-20	YPL	y/c coordinate of the point.
	21-30	ZPL	Spanwise coordinate of the point in case the section represents a nonplanar intersection between the wing and the fuselage or the wing and the nacelle.
			NACELLE INPUT
N1	1-80		Header.
N2	1-10	FNNAC	Number of nacelle sections to be input.
	11-20	FINA	Not active. Set to 0.
	21-30	FINB	Not active. Set to 0.
			NOTE: The nacelle section curves are similar to the fuselage section curves. The nacelle sections are to be input as sets of cards N3-N6. There must be FNI such sets for the inboard nacelle. The FNO section cuts for the outboard nacelle follow the inboard cuts.
N3	1-80		Header.
N4	1-10	XNAC	x coordinate at which the following section is located.
	11-20	FN	Number of points on that station.
N5	1-80		Header.
N6	1-10	XN	x coordinate.
	11-20	YN	y coordinate.
	21-30	ZN	z coordinate.

Card No.	Column No.	Name	Description
			GENERAL DATA
G1	1-80	TITLE	An 80-column title.
G2	1-80		Header card. A typical header card is shown below.
G3	1-10	FNX	Number of grid cells in the I direction in the computational domain in the coarsest mesh. In physical domain this number is equivalent to the number of cells in the chordwise direction of the wing and includes the cells on the entire surface of the wing as well as the wake.
	11-20	FNY	Number of grid cells in the J direction in the computational domain in the coarsest mesh. In the physical plane this corresponds to the number of cells in the direction normal to both the chord and the span.
	21-30	FNZ	Number of grid cells in the K direction in the computational domain in the coarsest mesh. In the physical plane this corresponds to the number of cells in the spanwise direction.
	31-40	FMESH	Number of meshes. In Euler analyses only the fine mesh grid is used. Hence the actual size of the mesh will be $FNX = FNX^{(FMESH-1)}$ etc.
	41-50	FSCR	I/O unit on which some of the intermediate results are written. If set to 0. then the program puts that information on TAPE13.
G2	1-80		Header card.
G5	1-10	FFUS	Parameter specifying the presence of the body geometry data in the input file. =0 if the body is absent. =1 if the body is present.

Card No.	Column No.	Name	Description
	11-20	FNAC	Parameter specifying the presence of the nacelle geometry data in the input file. =0 if the nacelle is absent. =1 if the nacelle is present.
	21-30	FWBIN	Parameter specifying the presence of the wing-body intersection geometry in the input file. =0 if the wing-body intersection is assumed to be flat. =1 if the wing-body intersection is to be input with the first wing section.
	31-40	FWNIN	Parameter specifying the presence of the nacelle-wing intersection geometry in the input file. =0 if the wing-nacelle intersection is assumed to be flat. =1 if the wing-nacelle intersection is to be input with the first wing section.
G6	1-80		Header card.
G7	1-10	FWDIS	Parameter specifying the spanwise distribution of the wing surface grid. =0 if the spanwise wing surface grid stations are to be placed at equal intervals. =1 if the spanwise wing surface grid stations are to be calculated using cosine rule. =2 if the spanwise stations in the wing surface grid are to be input by the user. User must input cards G8-G11 if FWDIS is 2.
	11-20	FSB	Factor used in the calculation of the cosine spacing. Used only if FWDIS=1.

Card No.	Column No.	Name	Description
	21-30	FST	Factor used in the calculation of the cosine spacing. Used only if FWDIS=1. NOTE: Input cards G8-G11 only if FWDIS=2.
G8	1-80		Header card.
G9	1-10	NSPAN	Number of spanwise grids. The user must input card G11 NSPAN times.
G10	1-80		Header card.
G11	1-10	ZSPAN	Spanwise location of the wing surface grid. This card must be input NSPAN times.
			WING DATA
W1	1-80		Header.
W2	1-10	FNS	Number of wing sections to be input.
	11-20	SWEEP	Leading edge sweep angle.
	21-30	DIHED	Dihedral angle. NOTE: Each wing section is input in its nondimensionalized form through the x/c and y/c coordinates referred to a coordinate system attached to the leading edge of the section. The upper and the lower surface of the section are input separately. In order to locate this section in the global coordinate system, the coordinate of the origin point (the leading edge) is also given in the global coordinate system along with certain scaling and pitching parameters. There must be FNS sets of cards W3-W10.
W3	1-80		Header.

Card No.	Column No.	Name	Description
W4	1-10	ZLE	z coordinate of the leading edge of the section to be input.
	11-20	XL	x coordinate of the leading edge of the section to be input.
	21-30	YL	y coordinate of the leading edge of the section to be input.
	31-40	CHORD	Chord length of the section being input.
	41-50	THICK	Thickness scaling for the section.
	51-60	AL	Pitching angle of the section. The airfoil is always pitched about the leading edge of the section. It is recommended that the user not use this feature of the input unless he is exactly sure of the effect of the rotation introduced because of this input.
	61-70	FSEC	Parameter to indicate whether the previous section is to be repeated. =0 new section on cards W5-W8 is expected. =1 previous section is to be used. NOTE: Cards W5-W10 are to be provided only if FSEC=0.
W5	1-80		Header.
W6	1-10	YSYM	Parameter indicating the symmetry of the section. =0 if the section is not symmetric. Both the upper and lower section curves are to be input. =1 if the section is symmetric. Only the upper section curve may be input. The lower section curve is derived by inverting the upper section curve about the origin and the x/c axis.
	11-20	FNU	Number of points on the upper section curve.

Card No.	Column No.	Name	Description
	21-30	FNL	Number of points on the lower section curve. NOTE: Note the FNL may be different from FNU only if YSYM is 0. If YSYM is 1. then the FNU and FNL must be equal.
W7	1-80		Header. NOTE: Card W8 must be repeated FNU times.
W8	1-10	XPU	x/c coordinate of the point on the upper section curve. The section must always start with the leading edge as the origin.
	11-20	YPU	y/c coordinate of the point.
	21-30	ZPU	Spanwise coordinate of the point in case the section represents a nonplanar intersection between the wing and the fuselage or the wing and the nacelle. NOTE: Cards W9-W10 are to be input only if YSYM=0.
W9	1-80		Header. NOTE: Card W10 must be repeated FNL times.
W10	1-10	XPL	x/c coordinate of the point on the lower section curve. The section must always start with the leading edge as the origin in the x/c and y/c coordinates.
	11-20	YPL	y/c coordinate of the point.
	21-30	ZPL	Spanwise coordinate of the point in case the section represents a nonplanar intersection between the wing and the fuselage or the wing and the nacelle.

B.2 PREPARATION OF FILES INPUT2

This file is required as an input to the grid generation program PRGRD2 used in Step 2. This file contains the information needed by the grid generation program besides the actual geometry information. The program requires input to control the iterative solution process, i.e. the number of iterations to be performed and the convergence tolerances to be used in the numerical method. The program also needs information such as the wing sweep and the dihedral angle to extend the flow grid beyond the wingtip in a smooth manner, the spacing in the farfield, and certain source control parameters that determine the grid distribution. The input format for this file is described next.

Card No.	Column No.	Name	Description
G1	1-80	TITLE	An 80-column title assigned to the run.
G2	1-80		Header card.
G3	1-10	FTEST	Grid generation parameter. Set to 3.
	11-20	FMESH	Number of meshes to be used in the grid generation. In the flow solution the mesh is refined (FMESH-1) times.
	21-30	FNSAV	Number of grids output.
	31-40	FMFRF	Not used.
	41-50	FPRINT	Set it to -1.
G4	1-80		Header card.
			NOTE: Read the following card FMESH times.
G5	1-10	FIT1	Number of iterations to be performed for 3D grid generation in this mesh. Recommended value 30.
	11-20	FIT2	Number of iterations to be performed for 2D grid generation at the root section in this mesh. Recommended value 30.
	21-30	FIT3	Number of iterations to be performed for 2D grid generation at the farfield section in this mesh. Recommended value 30.
	31-40	P10	Overrelaxation factor in 3D. Recommended value 1.25.
	41-50	P20	Overrelaxation factor in 2D at the root section. Recommended value 1.25.

Card No.	Column No.	Name	Description
			Recommended value 1.25.
	61-70	TOL	Minimum convergence error in residual of the solution allowed before the iterations are discontinued.
G6	1-80		Header.
G7	1-10	FSYM	Wing symmetry parameter. =0 if wing is not symmetric. =1 if the wing is symmetric.
	11-20	BODY	Parameter that determines the presence of a body. < 4 if there is no fuselage. =4 if the fuselage is present. =5 if the fuselage is present and the spanwise (y) coordinates of the wing root section (w/b intersection) are to be set to a constant value. > 6 if the fuselage is present and curved wing-fuselage intersection is desired.
	21-30	SWEEP	Identical to that in INPUT1a. Card W2.
	31-40	DIHED	Identical to that in INPUT1a. Card W2.
	41-50	FUS	Not used.
G8	1-80		Header card.
G9	1-10	DYFAC	Coarse mesh grid spacing in the normal direction normalized by the chord length.
	11-20	RFAC1	Ratio of the extent of the farfield in the normal (J) direction and the larger of the semispan or the fuselage body length at the wing body root or the first grid section.
	21-30	RFAC2	Ratio of the farfield in the normal (J) direction and the larger of the semispan or the fuselage body length at the last grid section at the spanwise farfield.

Card No.	Column No.	Name	Description
	31-40	AFAC	Ratio of the distance between the tip and the spanwise farfield and the larger of the semispan or the fuselage length.
	41-50	FREAD	I/O unit for wing geometry. Must be compatible with the filename containing the wing geometry. Set it to 12 to conform to the sample JCL file shown in the appendix.
	51-60	FRD2	I/O unit for fuselage geometry. Must be compatible with the filename containing the fuselage geometry. Set it to 8 to conform to the sample JCL file shown in the appendix.
	61-70	YFAC	
G10	1-80		Header card.
G11	1-10	FICKM	Set it to 1.
	11-20	FISCL	Source control parameter for 3D in I direction. Recommended value 1.
	21-30	FJSCL	Source control parameter for 3D in J direction. Recommended value 1.
	31-40	FKSCL	Source control parameter for 3D in K direction. Recommended value 1.
G12	1-80		Header card.
G13	1-10	FISCL2	Source control parameter for 2D at the root section in I direction. Recommended value 1.
	11-20	FJSCL2	Source control parameter for 2D at the root section in J direction. Recommended value .25 to 75.
	21-30	FISCL3	Source control parameter for 2D at the F-F section in K direction. Recommended value 1.
	31-40	FJSCL3	Source control parameter for 2D at the F-F section.
G14	1-80		Header card.

Card No.	Column No.	Name	Description
G15	1-10	FNAC	Nacelle control input. =6. if nacelle is present. =0 if nacelle is absent.
	11-20	FIRD3	I/O unit for nacelle geometry. Must be compatible with the filename containing the nacelle geometry. Set it to 4 to conform to the sample JCL file shown in the appendix.
	21-30	FPER	Parameter to change orientation of cutting plane for nacelle surface grids. < 0 if overwing. Recommended value -1. > 0 if underwing. Recommended value +1.
G16	1-80		Header card.
G17	1-10	FWDIS	Not used.
	11-20	FWALL	Parameter to specify the presence of endplates.

B.3 PREPARATION OF FILE INPUT3

This file is required by the postprocessor program (PRPOST) in Steps 3 and 5 and contains certain control parameters.

Card No.	Column No.	Name	Description
G1	1-80	TITLE	80-column title assigned to the run.
G2	1-80		Header card.
G3	1-10	FGRD	Parameter controlling the postprocessing of the volume grid. =0 if no grid processing is desired. =1 if grid planes and surface grid extraction are desired. This option should be used in Step 3.
	11-20	FFLOW	Parameter controlling the postprocessing of the flowfield. =0 if no flowfield processing is desired. =1 if aerodynamic information at flowfield planes is desired. This option should be used in Step 5.
	21-30	FGGP	Parameter controlling the postprocessing of the aerodynamic information on the surface. =0 if no surface information is desired. =1 if aerodynamic information at the surface is desired. This option should be used in Step 5. Read cards G4 and G5 only if either of the parameters FFLOW or FGGP on card G3 is nonzero.
G4	1-80		Header card.
G5	1-10	RM	Mach number.
G6	1-80		Header card.

Card No.	Column No.	Name	Description
G7	1-10	FNX	Number of grid cells in the I direction in the computational domain. In physical domain this number is equivalent to the number of cells in the chordwise direction of the wing and includes the cells on the entire surface of the wing as well as the wake.
	11-20	FNY	Number of grid cells in the J direction in the computational domain. In the physical plane this corresponds to the number of cells in the plane normal to both the chord and the span.
	21-30	FNZ	Number of grid cells in the K direction in the computational domain. In the physical plane this corresponds to the number of cells in the spanwise direction.
G8	1-80		Header card.
G9	1-10	FMNAC	Parameter specifying the presence of the nacelle in the configuration. =0 if no nacelle is present. =1 if nacelle is present in the configuration. Read cards G10-G17 only if the parameter FFLOW on card G3 is nonzero.
G10	1-80		Header card.
G11	1-10	FCUTI	Parameter controlling the output of the aerodynamic information on certain I planes in the flowfield. =0 if no I planes are desired. =0 if the I plane information is desired. The number input here will be the number of I planes that user wishes to extract from the flowfield. The I counters for the desired I planes are required to be input if this option is chosen.

Card No.	Column No.	Name	Description
	11-20	FCUTJ	Parameter controlling the output of the aerodynamic information on certain J planes in the flowfield. =0 if no J planes are desired. =0 if the J plane information is desired. The number input here will be the number of J planes that user wishes to extract from the flowfield. The J counters for the desired J planes are required to be input if this option is chosen.
	21-30	FCUTK	Parameter controlling the output of the aerodynamic information on certain K planes in the flowfield. =0 if no K planes are desired. =0 if the K plane information is desired. The number input here will be the number of K planes that user wishes to extract from the flowfield. The K counters for the desired K planes are required to be input if this option is chosen. Cards G12-G13 are required only if FCUTI is nonzero. If it is nonzero, then the number of G13 cards to be input must be exactly equal to FCUTI. The counter I in card G13 goes from 1 to FCUTI.
G12	1-80		Header card.
G13	1-10	ICUT(I)	I counter of the I plane that user wishes to extract.
	11-20	IJS(I)	J counter of the starting point in the I plane of interest. If this field is left blank or has a zero in it, the program will default to a value of 2 for this number.
	21-30	IJE(I)	J counter of the ending point in the I plane of interest. If this field is left blank or has a zero in it, the program will default to a value of NY+1 for this number.
	31-40	IKS(I)	K counter of the starting point in the I plane of interest. If this field is left blank or has a zero in it, the program will default to a value of 2 for this number.

Card No.	Column No.	Name	Description
	41-50	IKE(I)	K counter of the ending point in the I plane of interest. If this field is left blank or has a zero in it, the program will default to a value of NZ+1 for this number. Cards G14-G15 are required only if FCUTJ is nonzero. If it is nonzero, then the number of G15 cards to be input must be exactly equal to FCUTJ. The counter I in card G13 goes from 1 to FCUTJ.
G14	1-80		Header card.
G15	1-10	JCUT(I)	J counter of the J plane that user wishes to extract.
	11-20	JKS(I)	K counter of the starting point in the J plane of interest. If this field is left blank or has a zero in it, the program will default to a value of 2 for this number.
	21-30	JKE(I)	K counter of the ending point in the J plane of interest. If this field is left blank or has a zero in it, the program will default to a value of NZ+1 for this number.
	31-40	JIS(I)	I counter of the starting point in the J plane of interest. If this field is left blank or has a zero in it, the program will default to a value of 2 for this number.
	41-50	JIE(I)	I counter of the starting point in the J plane of interest. If this field is left blank or has a zero in it, the program will default to a value of NX+1 for this number. Cards G16-G17 are required only if FCUTK is nonzero. If it is nonzero, then the number of G17 cards to be input must be exactly equal to FCUTK. The counter I in card G17 goes from 1 to FCUTK.
G16			Header card.
G17	1-10	KCUT(I)	K counter of the K plane that user wishes to extract.
	11-20	KIS(I)	I counter of the starting point in the K plane of interest. If this field is left blank or has a zero in it, the program will default to a value of 2 for this number.
	21-30	KIE(I)	I counter of the starting point in the K plane of interest. If this field is left blank or has a zero in it, the program will default to a value of NX+1 for this number.

Card No.	Column No.	Name	Description
	31-40	KJS(I)	J counter of the starting point in the K plane of interest. If this field is left blank or has a zero in it, the program will default to a value of 2 for this number.
	41-50	KJE(I)	J counter of the ending point in the K plane of interest. If this field is left blank or has a zero in it, the program will default to a value of NY+1 for this number.
			Read cards G18-G25 only if the parameter FGRD on card G3 is nonzero.
G18			Header card.
G19	1-10	FNS	Parameter controlling the output of the surface grid information.
			=0 if no surface grid is desired.
			=0 if the surface grid coordinates are to be extracted.
	11-20	FNI	Parameter controlling the output of the I grid planes.
			=0 if no I planes are desired.
			=0 if the grid information for any I planes is desired. FNI is also the number of I planes the user wishes to extract.
	21-30	FNJ	Parameter controlling the output of the J grid planes.
			=0 if no J planes are desired.
			=0 if the grid information for any J planes is desired. FNJ is also the number of J planes the user wishes to extract.

Card No.	Column No.	Name	Description
	31-40	FNK	Parameter controlling the output of the K grid planes. =0 if no K planes are desired. =0 if the grid information for any K planes is desired. FNK is also the number of K planes the user wishes to extract. Cards G20-G21 are required only if FNI is nonzero. If it is nonzero, then the number of G21 cards to be input must be exactly equal to FNI. The counter I in card G21 goes from 1 to FNI.
G20			Header card.
G20	1-10	INET(I)	I counter of the I plane that user wishes to extract. Cards G22-G23 are required only if FNJ is nonzero. If it is nonzero, then the number of G23 cards to be input must be exactly equal to FNJ. The counter I in card G23 goes from 1 to FNJ.
G22			Header card.
G23	1-10	JNET(I)	J counter of the J plane that user wishes to extract. Cards G24-G25 are required only if FNK is nonzero. If it is nonzero, then the number of G25 cards to be input must be exactly equal to FNK. The counter I in card G25 goes from 1 to FNK.
G24			Header card.
G25	1-10	KNET(I)	K counter of the K plane that user wishes to extract.

B.4 PREPARATION OF FILE INPUT4

This file is required by the flow analysis program used in Steps 4a, 4b, and contains information regarding the flow condition and other controlling parameters. The differences in the file for Steps 4a and 4b are pointed out in the description.

Card No.	Column No.	Name	Description
G1	1-80	TITLE	80-column title assigned to the run.
G2	1-80		Header card.
G3	1-10	FNX	Number of grid cells in the I direction in the computational domain in the coarsest mesh. In physical domain this number is equivalent to the number of cells in the chordwise direction of the wing and includes the cells on the entire surface of the wing as well as the wake.
	11-20	FNY	Number of grid cells in the J direction in the computational domain in the coarsest mesh. In the physical plane this corresponds to the number of cells in the plane normal to both the chord and the span.
	21-30	FNZ	Number of grid cells in the K direction in the computational domain in the coarsest mesh. In the physical plane this corresponds to the number of cells in the spanwise direction.
	31-40	FMESH	Number of meshes. In Euler analyses only the final fine mesh is used. Hence the actual size of the mesh will be $FNX = FNX^{(FMESH-1)}$ etc.
	41-50	FCONT	Parameter that determines whether this run is a startup run or a restart run. =0 if it is startup run. Use this value in Step 4a. =1 if it is a restart run. This run works only with the finest mesh and the user is required to provide the fine grid file and the flowfield associated with the solution stored in a previous run. =2 if a combined coarse and fine mesh run is intended. This option will read in coarse mesh flowfield and refine it to calculate the fine mesh flowfield and perform a specified number of fine mesh interactions. With this option no additional coarse mesh interactions are performed.

Card No.	Column No.	Name	Description
G4	1-8		Header card. NOTE: Read the cards G5 FMESH times.
G5	1-10	NEND	Number of flow solution iterations to be performed in this mesh. Recommend value 300 in coarse and 300 in fine mesh.
	11-20	NPRNT	Interval at which the flow solution will be printed out. If NEND is smaller than NPRNT the program will print the solution at NEND cycles. If NEND is greater than NPRNT then the program will print the solution every NPRNT cycles and also at NENDth cycle.
	21-30	NOUT	Interval at which the convergence history information is written out.
	31-40	NTIM	Interval at which the local time step is calculated.
	41-50	IPRNT	Parameter used to control the frequency in I direction. The flowfield output is printed out at every Ith point. Recommmended value is 2 or 4.
	51-60	LPRNT	Parameter used to control the frequency of the K grid planes in the flowfield output. It is recommended that this number be 8 if there are 16 spanwise grids.
	61-70	SMP	Implicit smoothing control for each grid level. Set -1.8 for coarse grid, -2.6 for the fine grid.
G6			Header.
G7	1-10	CFL	CFL number. The CFL number is used in setting up the actual time step taken in the Runge-Kutta step. <0 if local time steps are to be used. Recommended values are -2. if SMOOP = 0 and -8. if SMOOP <0. A certain amount of experimentation may be required to determine the optimum values for CFL number. If the user is interested in the steady state solution then this option should be used since it will make the solution converge faster. >0 if the minimum time step is to be used to get a time-accurate solution.

Card No.	Column No.	Name	Description
	11-20	BCW	Parameter to control the manner in which the wing surface boundary condition extrapolation is performed. Recommended value is -1.
	21-30	QFIL	Parameter to control the frequency at which the filter terms are calculated. =0 if the filter terms are to be calculated only once during a Runge-Kutta step. =1 if the filter terms are to be calculated every stage of the Runge-Kutta step.
	31-40	VIS2	2nd order viscosity parameter. Recommended value is 1.
	41-50	VIS4	4th order viscosity parameter. Recommended value is .5.
	51-60	HM	Enthalpy damping term. This term is used in enhancing the convergence of a solution that is expected to have free stream total enthalpy everywhere. Recommended value is .05. This form of convergence acceleration should not be used in multienergy flows such as the propellers or exhaust flows.
	61-70	SMOOP	Not used.
	71-80	BCB	Body boundary condition extrapolation control.
G8	1-80		Header.
G9	1-10	FMACH	Mach number.
	11-20	ALPHA	Angle of attack.
G10			Header card.

Card No.	Column No.	Name	Description
G11	1-10	GRIDIN	Input grid control parameter. =0 grid is calculated internally. This option is effective for only the wing-alone configurations. If this option is selected the user must input the wing geometry as a part of this file in cards W1-W10. =1 grid is read in from a file.
	11-20	CC1	Parameter that weights the artificial viscosity term. Recommended value is .10.
	21-30	FMNAC	Parameter specifying the presence of the nacelle. =0 if the nacelle is absent. =1 if the nacelle is present.
	31-40	FWALL	Parameter indicating the presence of the sidewalls on the configuration. =0 if no sidewalls are present, the analysis is performed with one plane of symmetry. =1 if the sidewalls are present, the analysis is performed by applying the no-flow conditions at both ends of the grid.
G12			Header.
G13	1-10	REYNO	Reynolds number per unit of length in the configuration.
	11-20	TTOT	Total temperature of the ambient flow.
	21-30	PRNTL	Parameter controlling certain intermediate boundary layer output. Recommended value is 0. =0 if no printout is desired. =1 if additional printout is desired.

Card No.	Column No.	Name	Description
	31-40	XTU	x/c location of the transition on the upper surface (generally about .1).
	41-50	XTL	x/c location of the transition on the lower surface (generally about .1).
	51-60	CARTER	Relaxation parameter used in Carter algorithm. Recommended value is -1. 0. if Carter algorithm is to be used in the solution of the integral boundary layer equation. The absolute value of this number is used as a relaxation parameter. 0. if an algorithm developed by Larry Wigton is to be used in the solution of the integral boundary layer equation. The absolute value of this number is used as a relaxation parameter.
G14			Header.
G15	1-10	CFSTU	Skin friction coefficient to be used at the transition point defined by XTU in card 14 if no laminar flow calculations are to be performed.
	11-20	CFSTL	Skin friction coefficient to be used at the transition point defined by XTL in card 14 if no laminar flow calculations are to be performed.
	21-30	FLAGL	Parameter to indicate if the laminar flow calculations are to be performed until XTU and XTL defined in card G14.
G16			Header. NOTE: Cards G17 is to be read in FMESH Times.
G17	1-11	BLRN	Interval of the number of Euler iterations to be performed before a boundary layer calculation is made on the given mesh.
	11-20	PBLRN	Number of Euler iterations at which the boundary layer results are to be printed on that mesh.

Card No.	Column No.	Name	Description
G18			Header card.
G19	1-10	CUT	Number of flowfield cuts.
	1-80	XCUT	Nondimensional value of x station at which the cut is desired.
			WING DATA
			NOTE: The cards W1-W10 are to be input only if GRIDIN is 0.
W1	1-80		Header.
W2	1-10	FNS	Number of wing sections to be input.
	11-20	SWEEP	Leading edge sweep angle.
	21-30	DIHED	Dihedral angle.
			NOTE: Each wing section is input in this nondimensionalized form through the x/c and y/c coordinates referred to a coordinate system attached to the leading edge of the section. The upper and the lower surface of the section are input separately. In order to locate this section in the global coordinate system, the coordinate of the origin point (the leading edge) is also given in the global coordinate system along with certain scaling and pitching parameters. There must be FNS sets of cards W3-W10.
W3	1-80		Header.
W4	1-10	ZLE	z coordinate of the leading edge of the section to be input.
	11-20	XL	x coordinate of the leading edge of the section to be input.
	21-30	YL	y coordinate of the leading edge of the section to be input.
	31-40	CHORD	Chord length of the section being input.

Card No.	Column No.	Name	Description
	41-50	THICK	Thickness scaling for the section. The input airfoil y/c coordinate is multiplied by this factor to calculate the airfoil that is actually used in generating the surface grid. We do not recommend the use of this parameter. Set value to 1.
	51-60	AL	Pitching angle of the section. The airfoil is always pitched about the leading edge of the section. It is recommended that the user not use this feature of the input unless he is exactly sure of the effect of the rotation introduced because of this input.
	61-70	FSEC	Parameter to indicate whether the previous section is to be repeated. =0 new section on cards W5-W8 is expected. =1 previous section is to be used. NOTE: Cards W5-W10 are to be provided only if FSEC=0.
W5	1-80		Header.
W6	1-10	YSYM	Parameter indicating the symmetry of the section. =0 if the section is not symmetric. Both the upper and lower section curves are to be input. =1 if the section is symmetric. Only the upper section curve may be input. The lower section curve is derived by inverting the upper section curve about the origin and the x/c axis.
	11-20	FNU	Number of points on the upper section curve.

Card No.	Column No.	Name	Description
	21-30	FNL	Number of points on the lower section curve. NOTE: The FNL may be different from FNU only if YSYM is 0. If YSYM is 1. then the FUN and FNL must be equal.
W7	1-80		Header. NOTE: Card W8 must be repeated FNU times.
W8	1-10	XPU	x/c coordinate of the upper section curve. The section must always start with the leading edge as the origin.
	11-20	YPU	y/c coordinate of the point.
	21-30	ZPU	Spanwise coordinate of the point in case the section represents a nonplanar intersection between the wing and the fuselage or the wing and the nacelle. NOTE: Cards W9-W10 are to be input only if YSYM=0.
W9	1-80		Header. NOTE: Card W10 must be repeated FNL times.
W10	1-10	XPL	x/c coordinate of the lower section curve. The section must always start with the leading edge as the origin in the x/c and y/c coordinates.
	11-20	YPL	y/c coordinate of the point.
	21-30	ZPL	Spanwise coordinate of the point in case the section represents a nonplanar intersection between the wing and the fuselage or the wing and the nacelle.
P1			Header card.

Card No.	Column No.	Name	Description
P2	1-10	FMDSK	Parameter specifying the presence of the propeller.
			=0 No prop.
			=1 Tractor. User must input cards P3-P6.
			=2 Pusher. User must input cards P3-P6.
	11-20	FIRDSK	Number of loading input parameters.
P3			Header card.
P4	1-10	XDSKO	x coordinate of center of prop disk.
	11-20	YDSKO	y coordinate of center of prop disk.
	21-30	ZDSKO	z coordinate of center of prop disk.
	31-40	RDSK	Radius of the disk.
P5			Header.
P6	1-10	XDSK	Nondimensional r/R coordinate at which loading is defined.
	11-20	PSTG	Ratio of the total pressure aft of the propeller to the freestream static pressure.
	21-30	TSTF	Ratio of the total temperature aft of the propeller to the freestream static temperature.
	31-40	ALP	Swirl angle.
			>0 up inboard.
			<0 up outboard.

C.0 JOB CONTROL DECKS

This appendix presents the job control decks required in the execution of the system of programs. The various inputs required for the jobs are included after the control cards separated by an end of record mark. In addition, the program may require certain other data files. The job cards in the decks approximately indicate the computer resources required to run the job successfully.

C.1 JOB CONTROL DECK REQUIRED IN STEP 1A

This is a preprocessing job. It reads the user data file TAPE1 (described as file INPUT1a in Appendix B) and writes out fuselage curves on file TAPE2, wing curves on file TAPE3, and nacelle curves on file TAPE4 to be read by subsequent programs. This run does not require any input after the control cards. The job control cards are given below.

```
/JOB
/NOSEQ
Y11R,T100,CM50000.                              SAMANT
USER,USERNOC,PASSWOR.
CHARGE,CHRGNO,LRC.
GET,UP=TPGD11,DAT11=YWBN1.
UPDATE,P=0,I=UP,C=Y11CMP.
TOVPS(INPUT,C6UD=Y11CMP,DAT11,UN=USERNO,PW=CYBPWRD,AC=CHRGNO)
DAYFILE,Y11DAZ.
REPLACE,Y11DAZ.
EXIT.
DAYFILE,Y11DAZ.
REPLACE,Y11DAZ.
/EOR
Y11R.
USER,U=USERNO,PA=CYBPWRD,AC=CHRGNO.
RESOURCE,TL=200,LP=6,WS=832,JCAT=MDBAT.
DELIVER(SAMANT)
COMMENT. RECOVER PROGRAM AND DATA FILES.
ATTACH,Y11CMP,DAT11.
COMMENT. Y11CMP AND LOAD.
FORTRAN(I=Y11CMP,B=BIN/1000,L=Y11LST,OPT=B)
COPY,Y11LST,OUTPUT.
LOAD(BIN,CN=GO/1000,CDF=1000,GRLPALL= )
COMMENT. EXECUTE THE PROGRAM.
GO(TAPE1=DAT11,TAPE3=Y11W,TAPE6=Y11OUT)
COPY,Y11OUT,OUTPUT.
PURGE,Y11CMP,DAT11.
COMMENT. STORE FILES ON VSP.
PURGE,Y11W.
DEFINE,Y11W.
SUMMARY.
DAYFILE,Y11DAY.
TONOS(INPUT,C6UD=Y11DAY,Y11LST,Y11OUT,JCS="USERNOC","PASSWOR","CHRGNO")
TONOS(Z,C6UD=Y11W,JCS="USERNOC","PASSWOR","CHRGNO")
COMMENT. PROCESSING IF ERROR OCCURS.
EXIT.
COPY,Y11OUT,OUTPUT.
PURGE,Y11CMP,DAT11.
DAYFILE,Y11DAY.
TONOS(INPUT,C6UD=Y11DAY,Y11LST,Y11OUT,JCS="USERNOC","PASSWOR","CHRGNO")
TONOS(Z,C6UD=Y11W,JCS="USERNOC","PASSWOR","CHRGNO")
PATTACH,UTILITY.
IDUMP.
/EOR
```

C.2 JOB CONTROL DECK REQUIRED IN STEP 1B

This is a preprocessing job. It reads the user data file TAPE1 (described as file INPUT1b in Appendix B) and writes out wing curves file TAPE2 to be read by the volume grid generation program. This run does not require any input after the control cards. This run is made only when the nacelle is present in the configuration. The job control deck is given below.

```
/JOB
/NOSEQ
Y12R,T100,CM50000.                              SAMANT
USER,USERNOC,PASSWOR.
CHARGE,CHRGNO,LRC.
GET,UP=TPGD12,DAT12=YWBN2.
UPDATE,P=0,I=UP,C=Y12CMP.
TOVPS(INPUT,C6UD=Y12CMP,DAT12,UN=USERNO,PW=CYBPWRD,AC=CHRGNO)
DAYFILE,Y12DAZ.
REPLACE,Y12DAZ.
EXIT.
DAYFILE,Y12DAZ.
REPLACE,Y12DAZ.
/EOR
Y12R.
USER,U=USERNO,PA=CYBPWRD,AC=CHRGNO.
RESOURCE,TL=200,LP=6,WS=832,JCAT=MDBAT.
DELIVER(SAMANT)
COMMENT. RECOVER PROGRAM AND DATA FILES.
ATTACH,Y12CMP,DAT12.
COMMENT. Y12CMP AND LOAD.
FORTRAN(I=Y12CMP,B=BIN/1000,L=Y12LST,OPT=B)
COPY,Y12LST,OUTPUT.
LOAD(BIN,CN=GO/1000,CDF=1000,GRLPALL= )
COMMENT. EXECUTE THE PROGRAM.
GO(TAPE1=DAT12,TAPE3=Y12W,TAPE6=Y12OUT)
COPY,Y12OUT,OUTPUT.
PURGE,Y12CMP,DAT12.
COMMENT. STORE FILES ON VSP.
PURGE,Y12W.
DEFINE,Y12W.
SUMMARY.
DAYFILE,Y12DAY.
TONOS(INPUT,C6UD=Y12DAY,Y12LST,Y12OUT,JCS="USERNOC","PASSWOR","CHRGNO")
TONOS(Z,C6UD=Y12W,JCS="USERNOC","PASSWOR","CHRGNO")
COMMENT. PROCESSING IF ERROR OCCURS.
EXIT.
COPY,Y12OUT,OUTPUT.
PURGE,Y12CMP,DAT12.
DAYFILE,Y12DAY.
TONOS(INPUT,C6UD=Y12DAY,Y12LST,Y12OUT,JCS="USERNOC","PASSWOR","CHRGNO")
TONOS(Z,C6UD=Y12W,JCS="USERNOC","PASSWOR","CHRGNO")
PATTACH,UTILITY.
IDUMP.
/EOR
```

C.3 JOB CONTROL DECK REQUIRED IN STEP 1C

This is a preprocessing job. It reads the data files generated for the wing in the previous two steps and combines them into one file. The input files required are file TAPE1 (TAPE3 in Step 1a) and TAPE2 (TAPE3 in Step 1b). The program writes out a file containing the wing curves on file TAPE3 to be read by the volume grid generation program. This run does not require any input after the control cards. The job control cards are given below.

```
/JOB
/NOSEQ
Y13R,T100,CM50000.                              SAMANT
USER,USERNOC,PASSWOR.
CHARGE,CHRGNO,LRC.
GET,UP=PRGRD13.
UPDATE,P=0,I=UP,C=Y13CMP.
ATTACH,Y11W,Y12W.
TOVPS(INPUT,C6UD=Y13CMP,Y11W,Y12W,UN=USERNO,PW=CYBPWRD,AC=CHRGNO)
DAYFILE,Y13DAZ.
REPLACE,Y13DAZ.
EXIT.
DAYFILE,Y13DAZ.
REPLACE,Y13DAZ.
/EOR
Y13R.
USER,U=USERNO,PA=CYBPWRD,AC=CHRGNO.
RESOURCE,TL=200,LP=6,WS=832,JCAT=MDBAT.
DELIVER(SAMANT)
COMMENT. RECOVER PROGRAM AND DATA FILES.
COMMENT. Y13CMP AND LOAD.
ATTACH,Y13CMP.
FORTRAN(I=Y13CMP,B=BIN,L=Y13LST,OPT=BO)
PURGE,Y13CMP.
LOAD(BIN,CN=GO/1000,CDF=1000,GRLPALL= )
COMMENT. EXECUTE THE PROGRAM.
ATTACH,Y11W.
ATTACH,Y12W.
COPY,Y11W,TAPE1.
COPY,Y12W,TAPE2.
GO(TAPE3=Y13W,TAPE6=Y13OUT)
PURGE,Y11W.
PURGE,Y12W.
COMMENT. STORE FILES ON VSP.
DEFINE,Y13W.
FILES.
SUMMARY.
DAYFILE,Y13DAY.
TONOS(Z,C6UD=Y13LST,Y13OUT,Y13DAY,Y13W,JCS="USERNOC","PASSWOR","CHRGNO")
COMMENT. PROCESSING IF ERROR OCCURS.
EXIT.
PURGE,Y13CMP.
PURGE,Y11W.
PURGE,Y12W.
FILES.
SUMMARY.
DAYFILE,Y13DAY.
TONOS(Z,C6UD=Y13LST,Y13OUT,Y13DAY,Y13W,JCS="USERNOC","PASSWOR","CHRGNO")
PATTACH,UTILITY.
IDUMP.
/EOR
```

C.4 JOB CONTROL DECK REQUIRED IN STEP 2

This is a volume grid generation run. It reads the data files TAPE4 (TAPE4 generated in Step 1a), TAPE8 (TAPE2 generated in Step 1c or TAPE3 generated in Step 1a), TAPE12 (TAPE2 generated in Step 1a), and an input file (described as INPUT2 in Appendix B) and generates the volume grid file TAPE1. The job control cards and the input cards are given below.

```
/JOB
/NOSEQ
Y2R,T100,CM50000.                              SAMANT
USER,USERNOC,PASSWOR.
CHARGE,CHRGNO,LRC.
PURGE,Y2CMP/ST=LPF,NA.
DEFINE,Y2CMP.
GET,UP=TPGD2A.
UPDATE,P=0,I=UP,C=Y2CMP.
TOVPS(INPUT,C6UD=UN=USERNO,PW=CYBPWRD,AC=CHRGNO)
DAYFILE,Y2DAZ.
REPLACE,Y2DAZ.
EXIT.
DAYFILE,Y2DAZ.
REPLACE,Y2DAZ.
/EOR
Y2R.
USER,U=USERNO,PA=CYBPWRD,AC=CHRGNO.
RESOURCE,TL=200,LP=6,WS=832,JCAT=MDBAT.
DELIVER(SAMANT)
COMMENT. COMPILE Y2CMP AND LOAD.
MFLINK(Y2CMP,ST=MZZ,DD=C6,JCS="USER,USERNOC,PASSWOR."
      ,"CHARGE,CHRGNO,LRC.","ATTACH,Y2CMP.")
FORTRAN(I=Y2CMP,B=BIN/1000,L=Y2LST,OPT=BO)
RETURN,Y2CMP.
COPY,Y2LST,OUTPUT.
LOAD(BIN,CN=GO/1000,CDF=1000,GRLPALL= )
RETURN,BIN.
COMMENT. EXECUTE THE PROGRAM.
MFLINK(Y13W,ST=MZZ,DD=C6,JCS="USER,USERNOC,PASSWOR."
      ,"CHARGE,CHRGNO,LRC.","ATTACH,Y13W.")
MFLINK(Y11F,ST=MZZ,DD=C6,JCS="USER,USERNOC,PASSWOR."
      ,"CHARGE,CHRGNO,LRC.","ATTACH,Y11F.")
MFLINK(Y11N,ST=MZZ,DD=C6,JCS="USER,USERNOC,PASSWOR."
     ,"CHARGE,CHRGNO,LRC.","ATTACH,Y11N.")
COPY,Y13W,TAPE12.
COPY,Y11F,TAPE8.
COPY,Y11N,TAPE4.
GO(TAPE1=YGRC,TAPE2=YGRM,TAPE6=Y2OUT)
COPY,Y2OUT,OUTPUT.
COMMENT. STORE FILES ON WFS.
MFLINK(YGRC,ST=MZZ,DD=UU,JCS="USER,USERNOC,PASSWOR."
      ,"CHARGE,CHRGNO,LRC.","DEFINE,YGRC.")
MFLINK(YGRM,ST=MZZ,DD=UU,JCS="USER,USERNOC,PASSWOR."
      ,"CHARGE,CHRGNO,LRC.","DEFINE,YGRM.")
```

```
SUMMARY.
DAYFILE,Y2DAY.
TONOS(Z,C6UD=Y2DAY,Y2LST,Y2OUT,JCS="USERNOC","PASSWOR","CHRGNO")
COMMENT. PROCESSING IF ERROR OCCURS.
EXIT.
PURGE,Y2CMP.
DAYFILE,Y2DAY.
TONOS(Z,C6UD=Y2DAY,Y2LST,Y2OUT,JCS="USERNOC","PASSWOR","CHRGNO")
COMMENT. STORE FILES ON WFS.
MFLINK(YGRC,ST=MZZ,DD=UU,JCS="USER,USERNOC,PASSWOR."
      ,"CHARGE,CHRGNO,LRC.","DEFINE,YGRC.")
MFLINK(YGRM,ST=MZZ,DD=UU,JCS="USER,USERNOC,PASSWOR."
      ,"CHARGE,CHRGNO,LRC.","DEFINE,YGRM.")
PATTACH,UTILITY.
IDUMP.
/EOR
Y2. COARSE GRID. 64*8*12. NEW PROGRAMS.
FTEST     RFLM      FNSAV     FMMRF     FPPRINT
3.0       2.0       2.0       -1.0      1.0
FIT1      FIT2      FIT3      P1        P2        P3        TOL
30.0      30.0      30.0      1.30      1.30      1.30      0.001
30.0      30.0      30.0      1.30      1.30      1.30      0.001
FSYM      BODY      SWEPT     DIHL      FUS
2.00      6.        35.0911261 2.9554492 .25
DYFAC     RFAC1     RFAC2     AFAC      FREAD     FRD2      YFAC
0.10      4.0       2.5       2.5       12.0      8.0       2.0
FICKM     FISCL     FJSCL     FKSCL
1.00      1.00      1.00      1.00
FISCL2    FJSCL2    FISCL3    FJSCL3
1.0       0.75      1.0       1.0
FNAC      FIRD3     FPER
 6.0      4.0       -.013
FWDIS
1.
/EOR
```

C.5 JOB CONTROL DECK REQUIRED IN STEP 3 AND 5

This processes the volume grid file TAPE10 (TAPE1 generated in Step 2), the flowfield file TAPE2 (file TAPE1 generated in Step 4), and certain input data (described as file INPUT3 in Appendix B). Its primary purpose is to interrogate the grid and the flowfield to calculate the surface grid and the surface pressure and offbody flow properties. The job control cards and the input cards are given below.

```
/JOB
/NOSEQ
CFLGYPR,T100,CM50000.                              SAMANT
USER,USERNOC,PASSWOR.
CHARGE,CHRGNO,LRC.
TOVPS(INPUT,UN=USERNO,PW=CYBPWRD,AC=CHRGNO)
DAYFILE,CGYPSDZ.
REPLACE,CGYPSDZ.
EXIT.
DAYFILE,CGYPSDZ.
REPLACE,CGYPSDZ.
/EOR
CFLGYPR.
USER,U=USERNO,PA=CYBPWRD,AC=CHRGNO.
RESOURCE,TL=200,LP=15,WS=2000,JCAT=MDBAT.
DELIVER(SAMANT)
COMMENT. RECOVER PROGRAM AND DATA FILES,PR=*,L=FILES2.
MFLINK(PRPOST,ST=MZZ,DD=C6,JCS="USER,USERNOC,PASSWOR."
      ,"CHARGE,CHRGNO,LRC.","ATTACH,PRPOST.")
FORTRAN(I=PRPOST,B=BIN/1000,OPT=BO)
LOAD(BIN,CN=GO/1500,CDF=1500,GRLPALL= )
COMMENT. EXECUTE THE PROGRAM.
MFLINK(CFLGYF,ST=MZZ,DD=UU,JCS="USER,USERNOC,PASSWOR."
      ,"CHARGE,CHRGNO,LRC.","ATTACH,CFLGYF.")
MFLINK(YGRM,ST=MZZ,DD=UU,JCS="USER,USERNOC,PASSWOR."
      ,"CHARGE,CHRGNO,LRC.","ATTACH,YGRM.")
SWITCH(YGRM,RT=W)
SWITCH(CFLGYF,RT=W)
GO(TAPE2=CFLGYF,TAPE10=YGRM,TAPE6=CGYPSOT)
COPY,CGYPSOT,OUTPUT.
COPY,TAPE31,CGYGGPW.
COPY,TAPE70,YNETS.
COPY,TAPE71,YNETI.
COPY,TAPE72,YNETJ.
COPY,TAPE73,YNETK.
COPY,TAPE81,CGYFLOI.
SUMMARY.
DAYFILE,CGYPSDY.
TONOS(Z,C6UD=CGYGGPW,CGYFLOI,JCS="USERNOC","PASSWOR","CHRGNO")
TONOS(Z,C6UD=YNETS,YNETI,YNETJ,YNETK,JCS="USERNOC","PASSWOR","CHRGNO")
COMMENT. PROCESSING IF ERROR OCCURS.
EXIT.
COPY,CGYPSOT,OUTPUT.
COPY,TAPE70,YNETS.
```

```
SUMMARY.
DAYFILE,CGYPSDY.
DAYFILE,CGYPSDY.
TONOS(Z,C6UD=CGYPSDY,CGYPSOT,JCS="USERNOC","PASSWOR","CHRGNO")
TONOS(Z,C6UD=CGYGGPW,CGYFLOI,JCS="USERNOC","PASSWOR","CHRGNO")
TONOS(Z,C6UD=YNETS,YNETI,YNETJ,JCS="USERNOC","PASSWOR","CHRGNO")
PATTACH,UTILITY.
IDUMP.
/EOR
GY : GRID INSPECTION AND FLOW POST PROCESSING.
FGRD        FFLOW       FGGP
1.          1.          1.00
RM
.8
FNX         FNY         FNZ
64.0        8.          12.0
FMNAC
1.
FLOWI
1.
FPLANEI
108.
FNS         FNI         FNJ         FNK
1.          1.          2.
INET
33.
JNET
1.
9.
/EOR
```

C.6 JOB CONTROL DECK REQUIRED IN STEP 4

This is the flow analysis run. This program requires a grid file TAPE10 (TAPE1 generated in Step 2) and input data (INPUT4a described in Appendix B). This run generates a convergence history file TAPE23, a loading file TAPE24 and the flowfield file TAPE1. It is recommended that the flowfield variable file TAPE1 be saved after every run. This file can be used to restart the next flow analysis run. The job control cards and the input cards described in INPUT4a are given below.

```
/JOB
/NOSEQ
CFLGYR,T100,CM50000.                                SAMANT
USER,USERNOC,PASSWOR.
CHARGE,CHRGNO,LRC.
TOVPS(INPUT,UN=USERNO,PW=CYBPWRD,AC=CHRGNO)
DAYFILE,CFLGYDZ.
REPLACE,CFLGYDZ.
EXIT.
DAYFILE,CFLGYDZ.
REPLACE,CFLGYDZ.
/EOR
CFLGYR.
USER,U=USERNO,PA=CYBPWRD,AC=CHRGNO.
RESOURCE,TL=500,LP=40,WS=5500,JCAT=MDBAT.
DELIVER(SAMANT)
MFLINK(AFLGBIN,ST=MZZ,DD=UU,JCS="USER,USERNOC,PASSWOR."
      ,"CHARGE,CHRGNO,LRC.","ATTACH,AFLGBIN.")
LOAD(AFLGBIN,CN=GO/1500,CDF=1500,GRLPALL= )
RETURN,AFLGBIN.
COMMENT. EXECUTE THE PROGRAM.
COPY,INPUT,DATAA.
COPY,DATAA,OUTPUT.
COPY,INPUT,DATAB.
COPY,DATAB,OUTPUT.
MFLINK(YGRM,ST=MZZ,DD=UU,JCS="USER,USERNOC,PASSWOR."
      ,"CHARGE,CHRGNO,LRC.","ATTACH,YGRM.")
SWITCH(YGRM,RT=W)
REQUEST(AFLGYF/500,RT=W)
GO(TAPE5=DATAA,TAPE10=YGRM,TAPE1=AFLGYF,TAPE6=CFLGYOA)
COPY,CFLGYOA,OUTPUT.
COMMENT. STORE FILES ON VSP.
MFLINK(CFLGYOA,ST=MZZ,DD=C6,JCS="USER,USERNOC,PASSWOR."
      ,"CHARGE,CHRGNO,LRC.","PURGE,CFLGYOA/NA.","DEFINE,CFLGYOA.")
SUMMARY.
MFLINK(BFLGBIN,ST=MZZ,DD=UU,JCS="USER,USERNOC,PASSWOR."
      ,"CHARGE,CHRGNO,LRC.","ATTACH,BFLGBIN.")
LOAD(BFLGBIN,CN=GO/5000,CDF=5000,GRLPALL= )
RETURN,BFLGBIN.
COMMENT. EXECUTE THE PROGRAM.
REQUEST(CFLGYF/2000,RT=W)
GO(TAPE2=AFLGYF,TAPE5=DATAB,TAPE10=YGRM,TAPE1=CFLGYF,TAPE6=CFLGYOB)
COPY,CFLGYOB,OUTPUT.
COMMENT. STORE FILES ON VSP.
MFLINK(CFLGYOB,ST=MZZ,DD=C6,JCS="USER,USERNOC,PASSWOR."
      ,"CHARGE,CHRGNO,LRC.","PURGE,CFLGYOB/NA.","DEFINE,CFLGYOB.")
MFLINK(CFLGYF,ST=MZZ,DD=UU,JCS="USER,USERNOC,PASSWOR."
      ,"CHARGE,CHRGNO,LRC.","PURGE,CFLGYF/NA.","DEFINE,CFLGYF.")
```

```
SUMMARY.
DAYFILE,CFLGYDY.
TONOS(Z,C6UD=CFLGYDY,JCS="USERNOC","PASSWOR","CHRGNO")
EXIT.
COMMENT. PROCESSING IF ERROR OCCURS.
COPY,CFLGYOA,OUTPUT.
COPY,CFLGYOB,OUTPUT.
MFLINK(CFLGYOA,ST=MZZ,DD=C6,JCS="USER,USERNOC,PASSWOR."
      ,"CHARGE,CHRGNO,LRC.","PURGE,CFLGYOA/NA.","DEFINE,CFLGYOA.")
MFLINK(CFLGYOB,ST=MZZ,DD=C6,JCS="USER,USERNOC,PASSWOR."
      ,"CHARGE,CHRGNO,LRC.","PURGE,CFLGYOB/NA.","DEFINE,CFLGYOB.")
MFLINK(CFLGYF,ST=MZZ,DD=UU,JCS="USER,USERNOC,PASSWOR."
      ,"CHARGE,CHRGNO,LRC.","PURGE,CFLGYF/NA.","DEFINE,CFLGYF.")
SUMMARY.
DAYFILE,CFLGYDY.
TONOS(Z,C6UD=CFLGYDY,JCS="USERNOC","PASSWOR","CHRGNO")
EXIT.
SUMMARY.
DAYFILE,CFLGYDY.
TONOS(Z,C6UD=CFLGYDY,JCS="USERNOC","PASSWOR","CHRGNO")
/EOR
CFLGY. PROP OFF
NX          NY          NZ          MMESH       FCONT
 64.         8.          12.          1.          0.
NEND        NPRNT       NOUT        NTIM        IPRNT       LPRNT       SMOVPV
300.         300.        10.         20.         0.          4.          -1.8
CFL         BCW         Q FIL       VIS 2       VIS 4       H FACTOR    SMOOP       BCB
-8.0        -1.0         0.          1.          0.5          .05         -2.6        -1.
FMACH       ALPHA
 .800        1.84
GRIDIN      CC1         FMNAC
2.0         0.10        2.0
FNCUT
4.
XCUT
1.00        2.00        5.00        10.0
COUPLING    NBC
0.          0.
FMDSK       FIRDSK
0.          21.0
XDSK0       YDSK0       ZDSK0       RDSK
90.0 0      -3.3        42.56553    10.0
RDS         PSTG        TSTG        AL1
0.0         1.52434     1.12800     0.0
3.0         1.52434     1.12800     0.0
3.25        1.52434     1.12800     0.0
3.3         1.52663     1.12848     0.0
3.5         1.53653     1.13057     3.6
4.0         1.55940     1.13535     7.2
4.5         1.57998     1.13961     8.7
5.0         1.59675     1.14305     9.4
5.5         1.61047     1.14585     9.5
6.0         1.62190     1.14817     9.4
6.5         1.63104     1.15001     9.2
7.0         1.64019     1.15185     8.8
7.5         1.64598     1.15301     8.58
8.          1.64705     1.15322     8.83
8.35        1.64751     1.15332     8.9
8.5         1.64781     1.15338     8.85
9.0         1.63867     1.15155     8.4
9.3         1.62037     1.14786     7.7
9.5         1.55483     1.13440     7.0
9.68        1.52891     1.12896     6.0
10.0        1.52434     1.12800     3.2
END OF CALCULATION
NX          NY          NZ
```

```
 0.          0.          0.
/EOR
CFLGY. PROP OFF
NX          NY          NZ          MMESH       FCONT
 64.        8.          12.          2.          2.00
NEND        NPRNT       NOUT        NTIM        IPRNT       LPRNT       SMOVPV
0.00         0.00        10.         20.         0.          4.          -1.8
100.         100.        10.         10.0        0.          4.          -2.6
CFL         BCW         Q FIL       VIS 2       VIS 4       H FACTOR    SMOOP       BCB
-8.0        -1.0         0.          1.          0.5          .05         -2.6        -1.
FMACH       ALPHA
 .800        1.84
GRIDIN      CC1         FMNAC
2.0         0.10        2.0
FNCUT
4.
XCUT
1.00        2.00        5.00        10.0
COUPLING    NBC
0.          0.
FMDSK       FIRDSK
0.          21.0
XDSK0       YDSK0       ZDSK0       RDSK
90.0 0      -3.3        42.56553    10.0
RDS         PSTG        TSTG        ALl
0.0         1.52434     1.12800     0.0
3.0         1.52434     1.12800     0.0
3.25        1.52434     1.12800     0.0
3.3         1.52663     1.12848     0.0
3.5         1.53653     1.13057     3.6
4.0         1.55940     1.13535     7.2
4.5         1.57998     1.13961     8.7
5.0         1.59675     1.14305     9.4
5.5         1.61047     1.14585     9.5
6.0         1.62190     1.14817     9.4
6.5         1.63104     1.15001     9.2
7.0         1.64019     1.15185     8.8
7.5         1.64598     1.15301     8.58
8.          1.64705     1.15322     8.83
8.35        1.64751     1.15332     8.9
8.5         1.64781     1.15338     8.85
9.0         1.63867     1.15155     8.4
9.3         1.62037     1.14786     7.7
9.5         1.55483     1.13440     7.0
9.68        1.52891     1.12896     6.0
10.0        1.52434     1.12800     3.2
END OF CALCULATION
NX          NY          NZ
 0.          0.          0.
/EOR
```

Mesh size	Number of iterations	Boeing CRAY	CYBER 203 scalar	CYBER 205 scalar	CYBER 205 vector
64 x 8 x 12	10	2.5	36 unoptimized	8.4 optimized	1.78
64 x 8 x 12	100	25	—	83.1	17.8
128 x 16 x 24	10	17	—	64.32	10.3

Table 1. Timing Data for Scalar and Vector Versions of the Propfan Euler Code

Case	Run	Geometry	Prop	B-layer	Grid size	Iterations	CPU
1	1	W/B/N	No	No	128 x 16 x 24	500	560
	2	W/B/N	Tractor	No	128 x 16 x 24	500	560
	3	W/B/N	Pusher	No	128 x 16 x 24	500	560
2	4	W alone	No	No	128 x 16 x 24	300	340
	5	W alone	No	Yes	128 x 16 x 24	300	740*
3	6	W/N	No	No	128 x 16 x 24	300	390
	7	W/N	Tractor	No	128 x 16 x 24	300	390

*Includes stripwise boundary layer coupling

Table 2. Summary of Test Cases Run on the Euler Code

Step	Program	Input	Output	Purpose
1a.	PRGRD11	TAPE1 (user data)	TAPE2 (body) TAPE3 (inboard wing) TAPE4 (nacelle)	Generate surface lines for use in steps 1c and 2
1b.	PRGRD12	TAPE1 (user data)	TAPE3 (outboard wing)	Generate surface lines for use in step 1c
1c.	PRGRD13	TAPE1 (TAPE3 step 1a) TAPE2 (TAPE3 step 2b)	TAPE3 (wing)	Combine wing data files
2.	PRGRD2A	TAPE12 (TAPE4 step 1a) TAPE4 (TAPE2 step 1a) TAPE8 (TAPE3 step 1c) INPUT2	TAPE2 (fine grid)	Generate flow grid
3.	PRPOST	TAPE10 (TAPE2 in step 2) INPUT3	TAPE70 (surface network) TAPE71 (I-planes) TAPE72 (J-planes) TAPE73 (K-planes)	Surface grid Volume grid Grid planes
4.	PRFLOW	TAPE10 (TAPE2 step 2) INPUT4	TAPE1 (flowfield) TAPE23 (convergence history) TAPE24 (wing load)	Obtain flow Solution for w/b/n
5.	PRPOST	TAPE10 (TAPE2 in step 2) TAPE2 (TAPE1 in step 4) INPUT3	TAPE31 (wing C_p) TAPE32 (body C_p) TAPE81 (C_p I planes) TAPE82 (C_p J planes) TAPE83 (C_p K planes)	Surface C_p Body C_p Flow C_p etc.

Table A.1. The Files Used for Data Transfer Among Various Segments of the Program

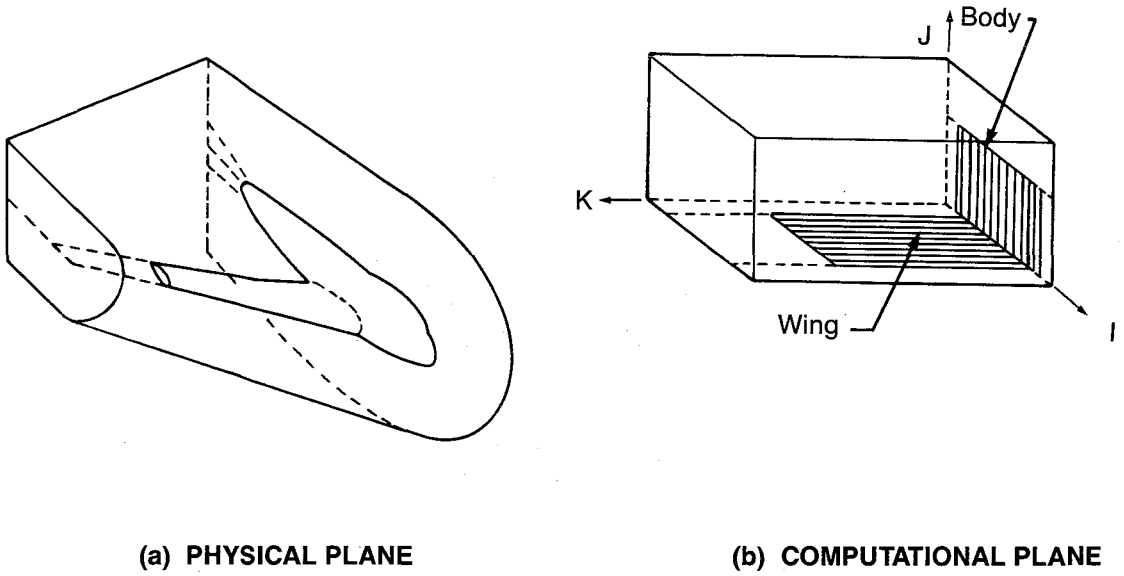


Figure 1. The Flow Grid System in the Physical and Computational Domain for Wing-Body Case

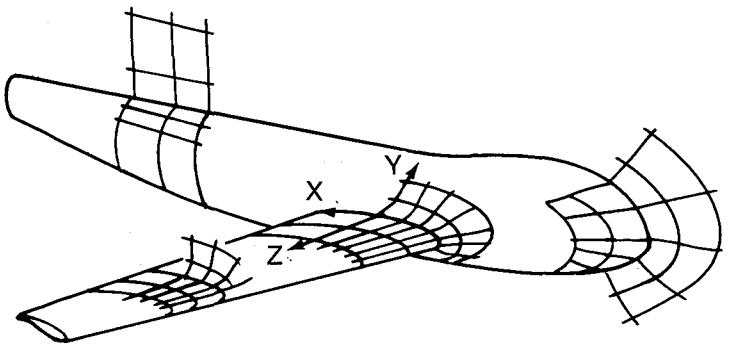
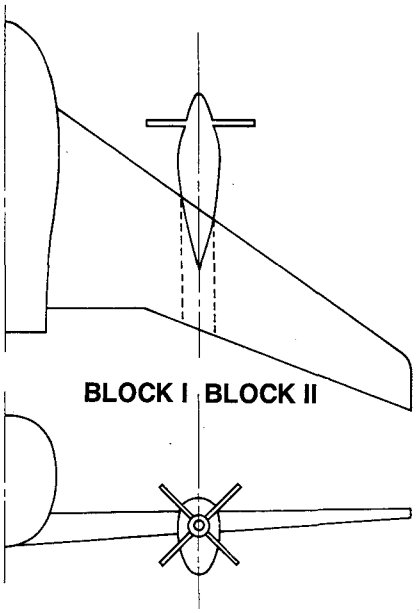
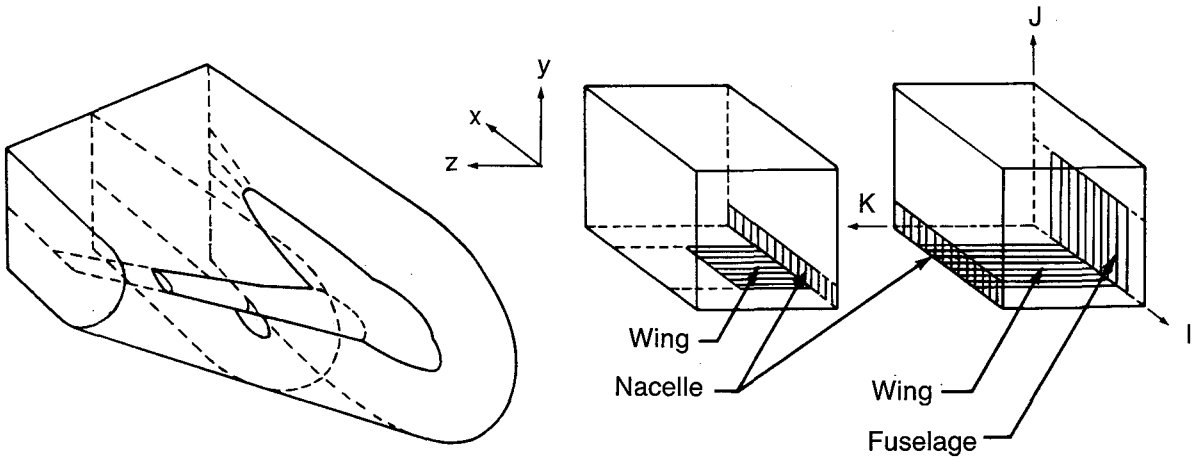


Figure 2. Surface Grid Structure



(a) DIVISION OF FLOW VOLUME INTO BLOCKS



(b) PHYSICAL PLANE AND COMPUTATIONAL PLANE

Figure 3. The Flow Grid System in the Physical and Computational Domain for Wing/Body/Nacelle

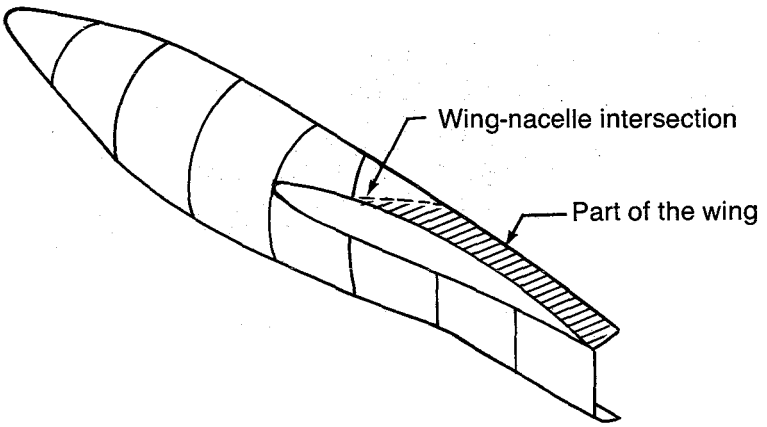
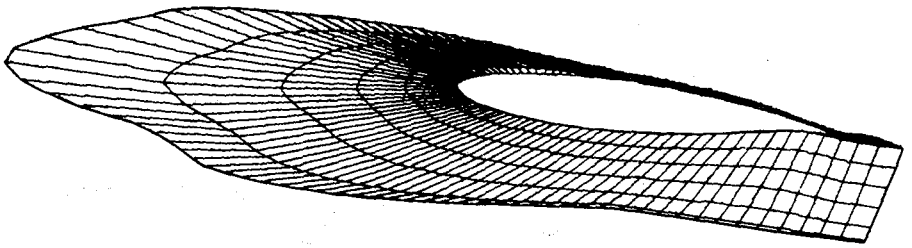
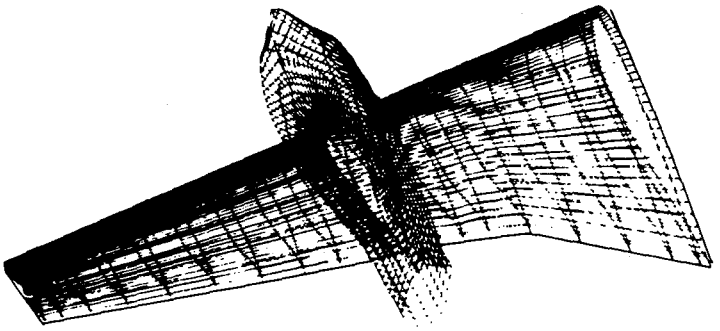


Figure 4. Redefinition of Nacelle Surface

SURFACE GRID ON THE FUSELAGE



SURFACE GRID ON THE NACELLE



SURFACE GRID ON WING-NACELLE

Figure 5. Surface Grid for a Propfan Configuration

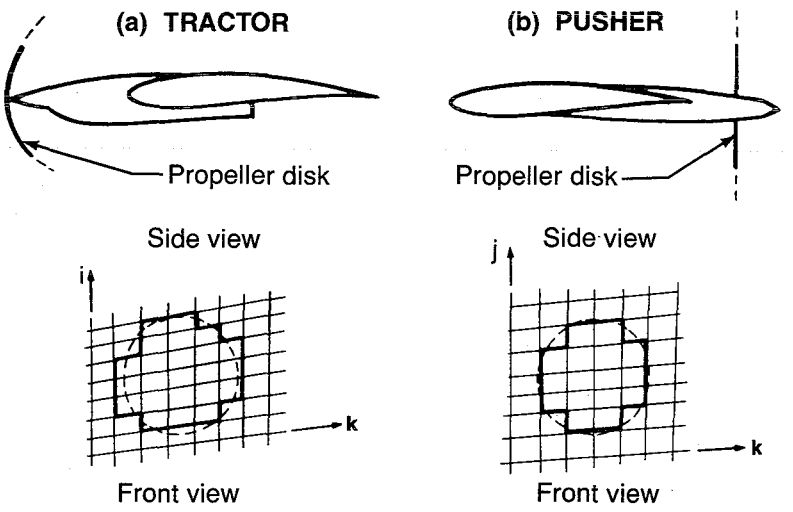
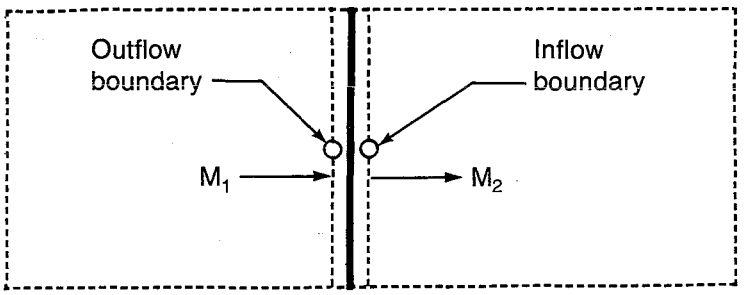


Figure 6. Representation of Propeller Disk for a Wing-Mounted Configuration



Case 1. $M_1, M_2, < 1$

Downstream of disk: Inflow boundary.
Extract q_2 from the solution. Specify the rest of the flow variables.

Upstream of disk: Outflow boundary.
Require continuity of mass flux, ϱq_n, across the disk. Extract the other flow variables from the solution.

Case 2. $M_1 > 1, M_2 < 1$

Downstream of disk: Inflow boundary.
Extract q_2 from the solution. Require continuity of ϱq_n, across the disk. Specify the rest of the flow variables.

Upstream of disk: Outflow boundary.
No boundary condition. Extract all flow variables from the solution.

Figure 7. Boundary Conditions Along a Propeller Disk

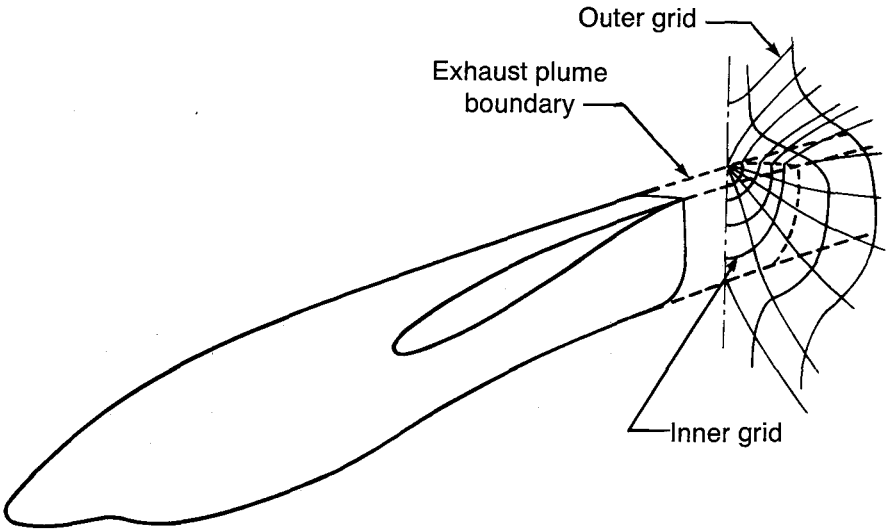


Figure 8. Illustration of Exhaust Plume Grid Embedding

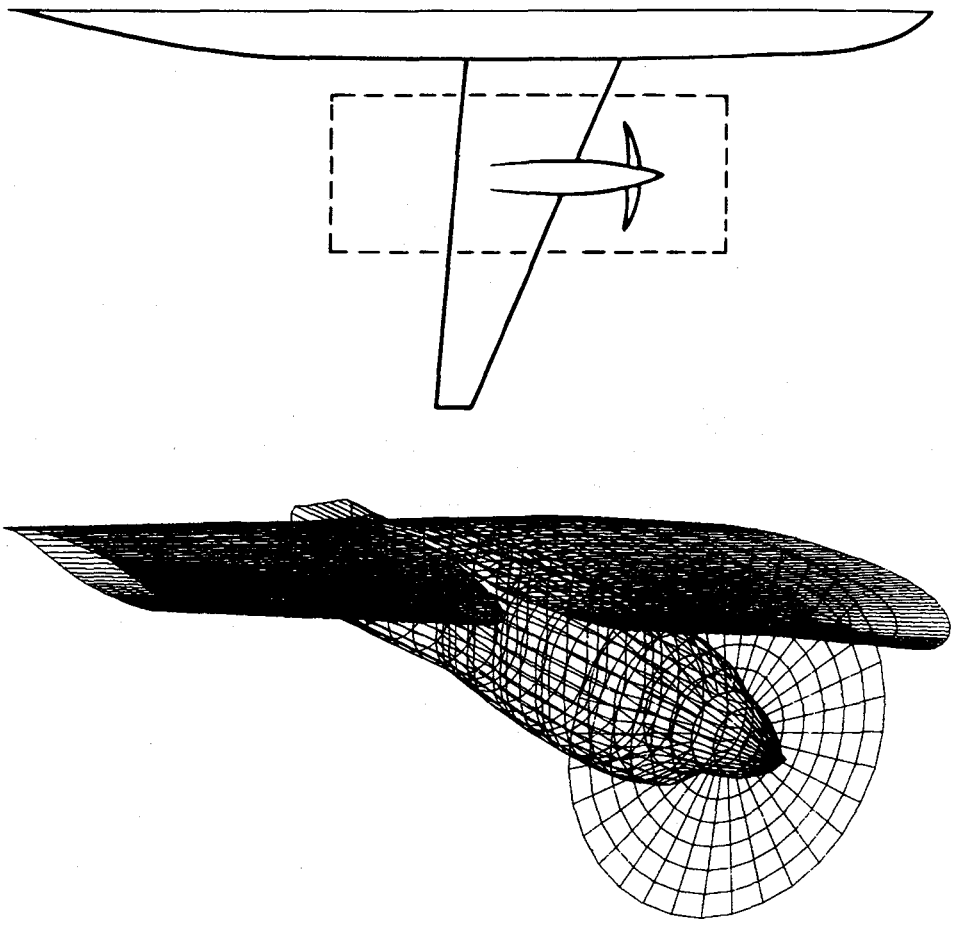


Figure 9. Embedded Nacelle-Propeller-Exhaust Plume Region

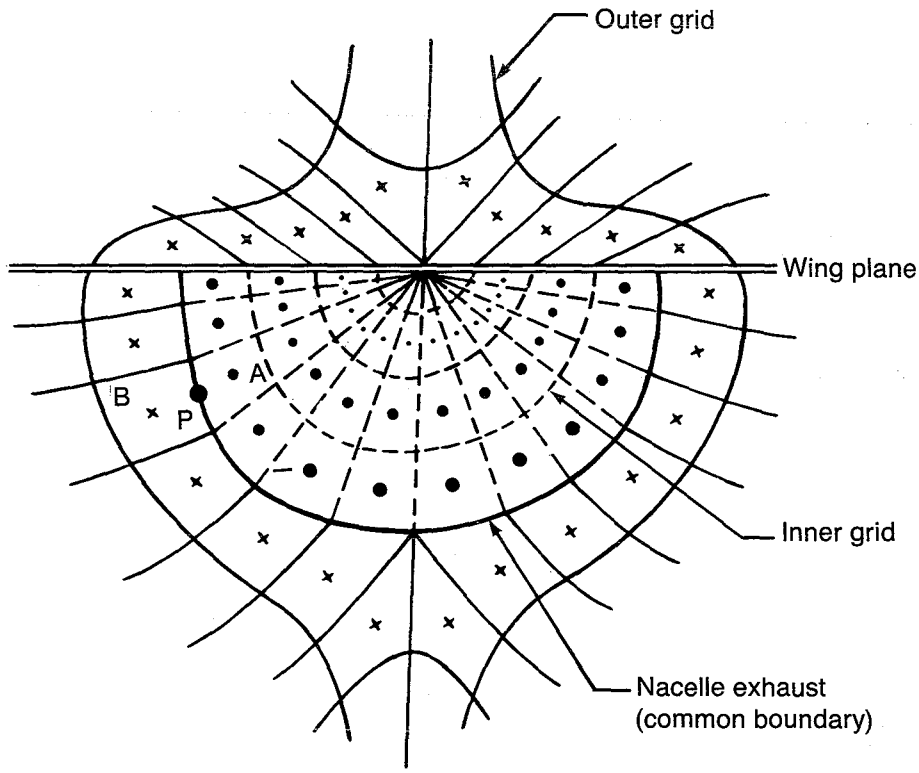


Figure 10. Exhaust Flow Grid Embedding

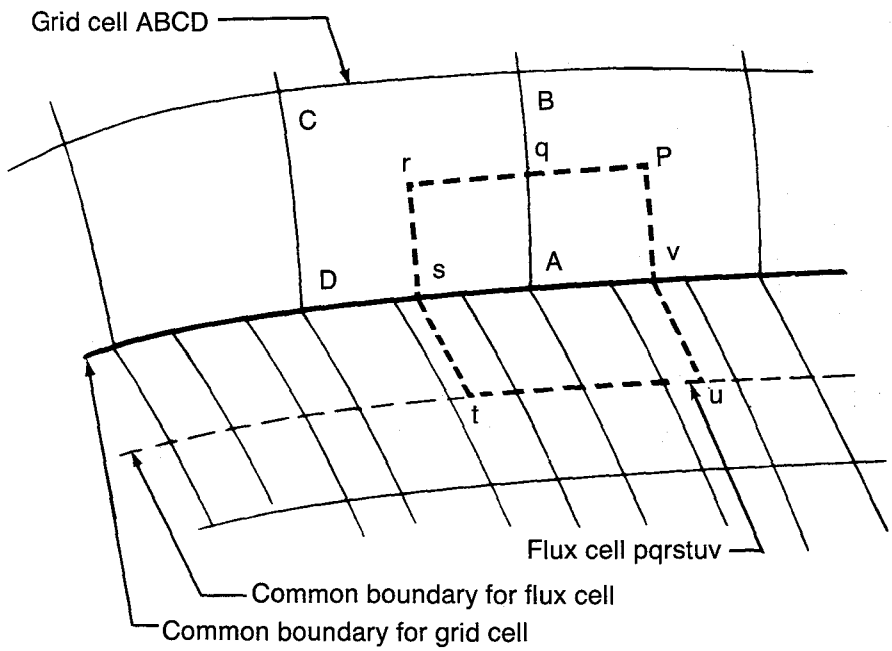
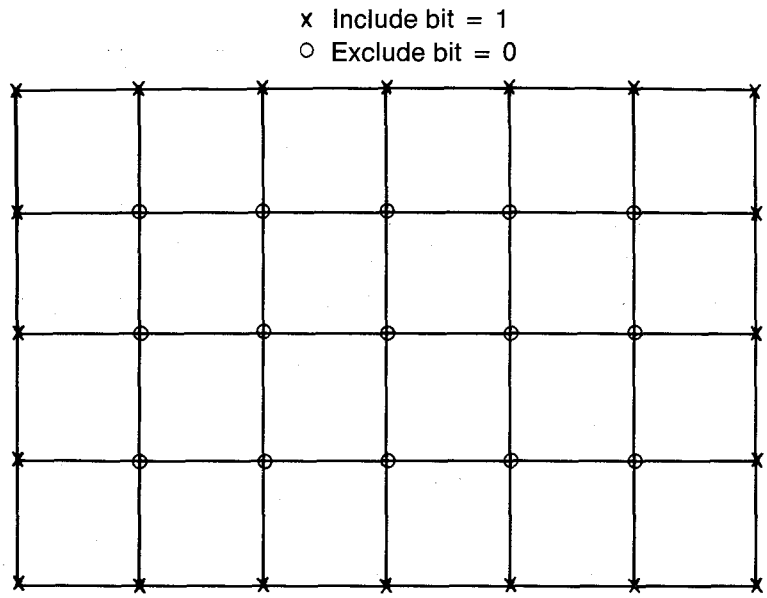
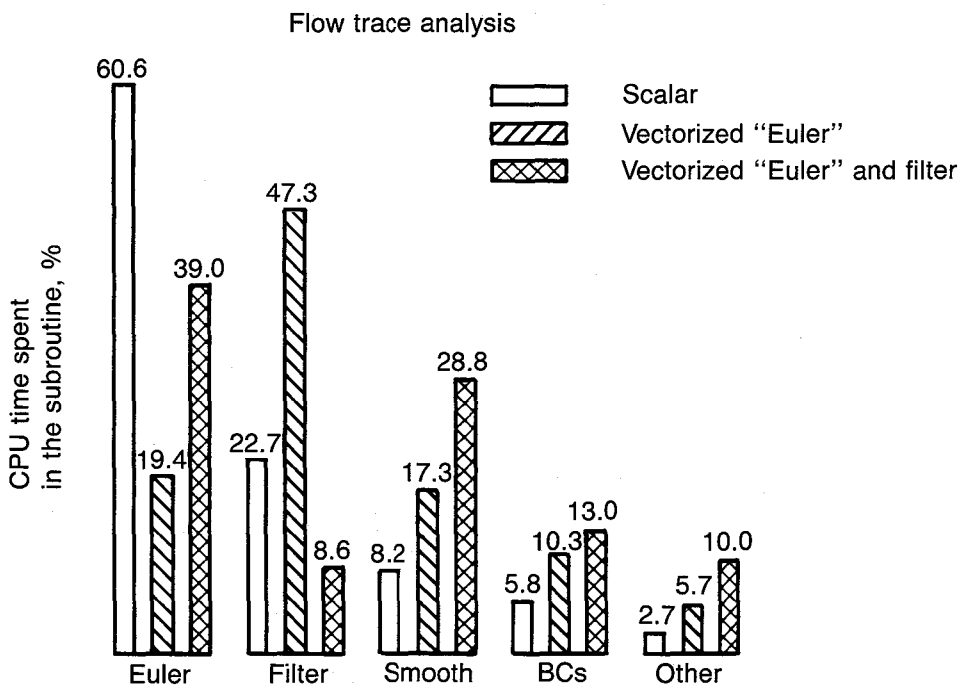


Figure 11. Nodal Point Scheme at the Common Boundary



(a) MASKING BIT PATTERN IN 2D VECTORIZATION



(b) EFFECT OF VECTORIZATION ON CPU TIME

Figure 12. Flow Code Vectorization Issues

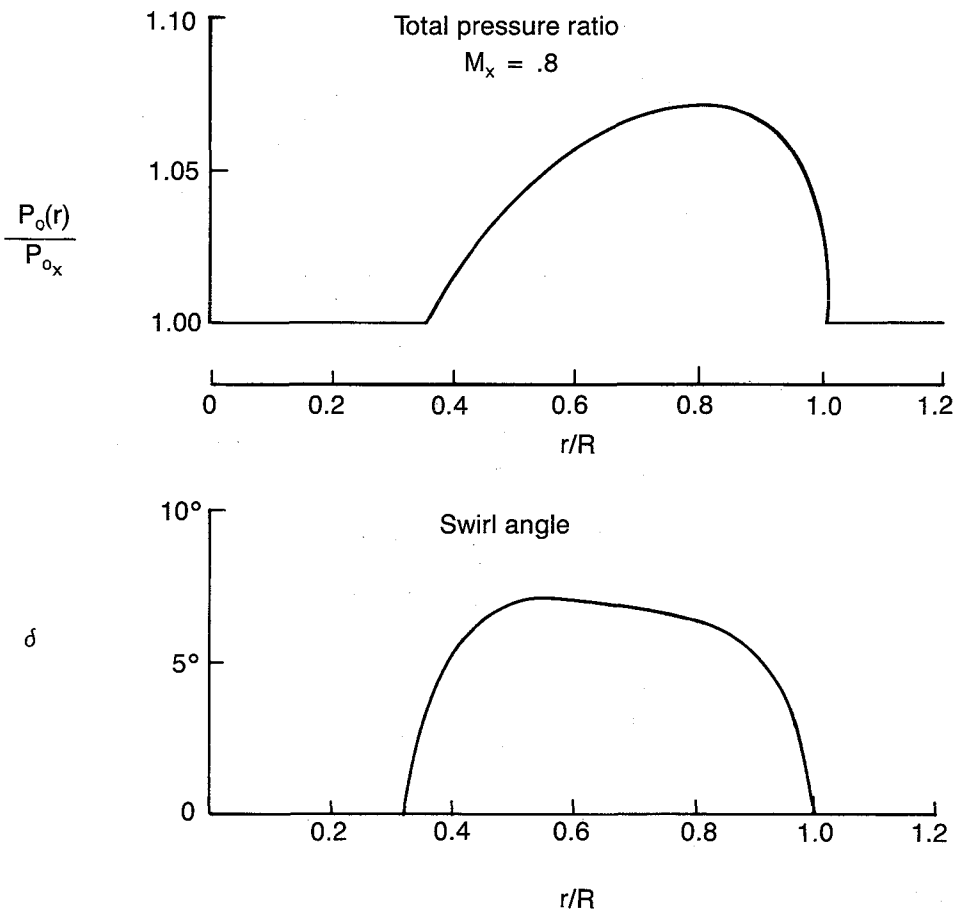


Figure 13. Radial Distribution of Swirl and Total Pressure in Prop Slipstream

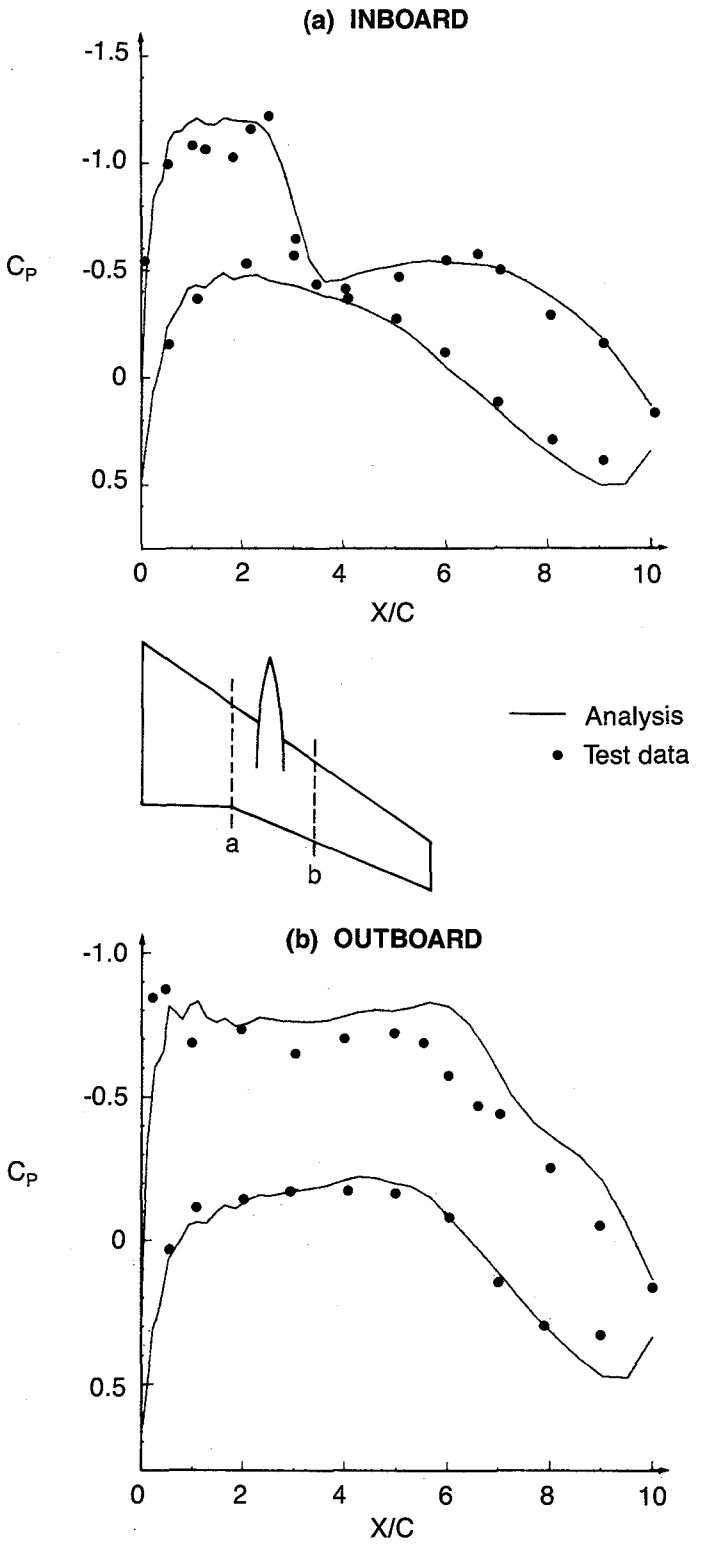


Figure 14. Comparisons of Wing Surface Pressures for a Turboprop Configuration With Propeller Off at $M_\infty = .80$, $\alpha = 1.84$

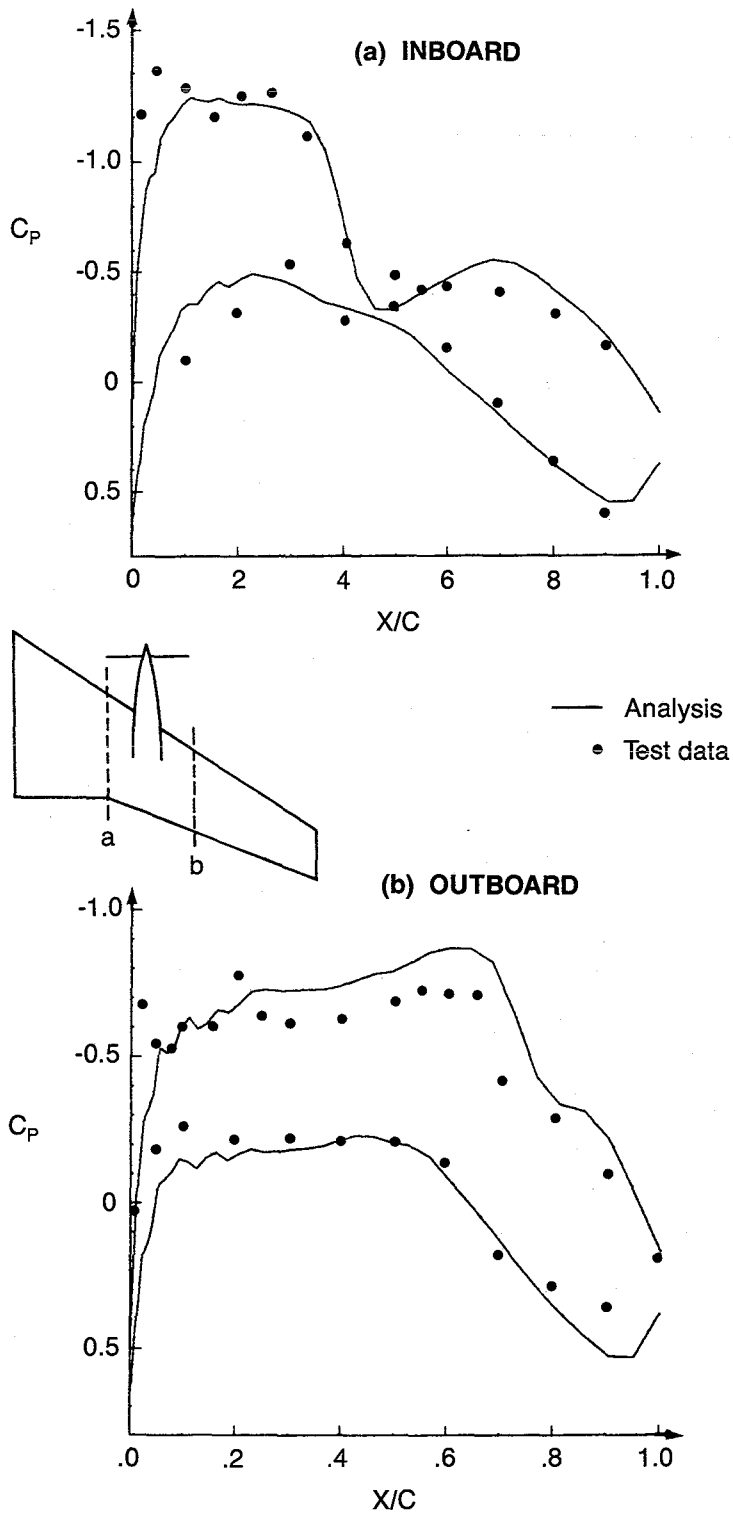


Figure 15. Comparisons of Wing Surface Pressures for a Turboprop Configuration With Propeller On at $M_\infty = .80$, $\alpha = 1.84$

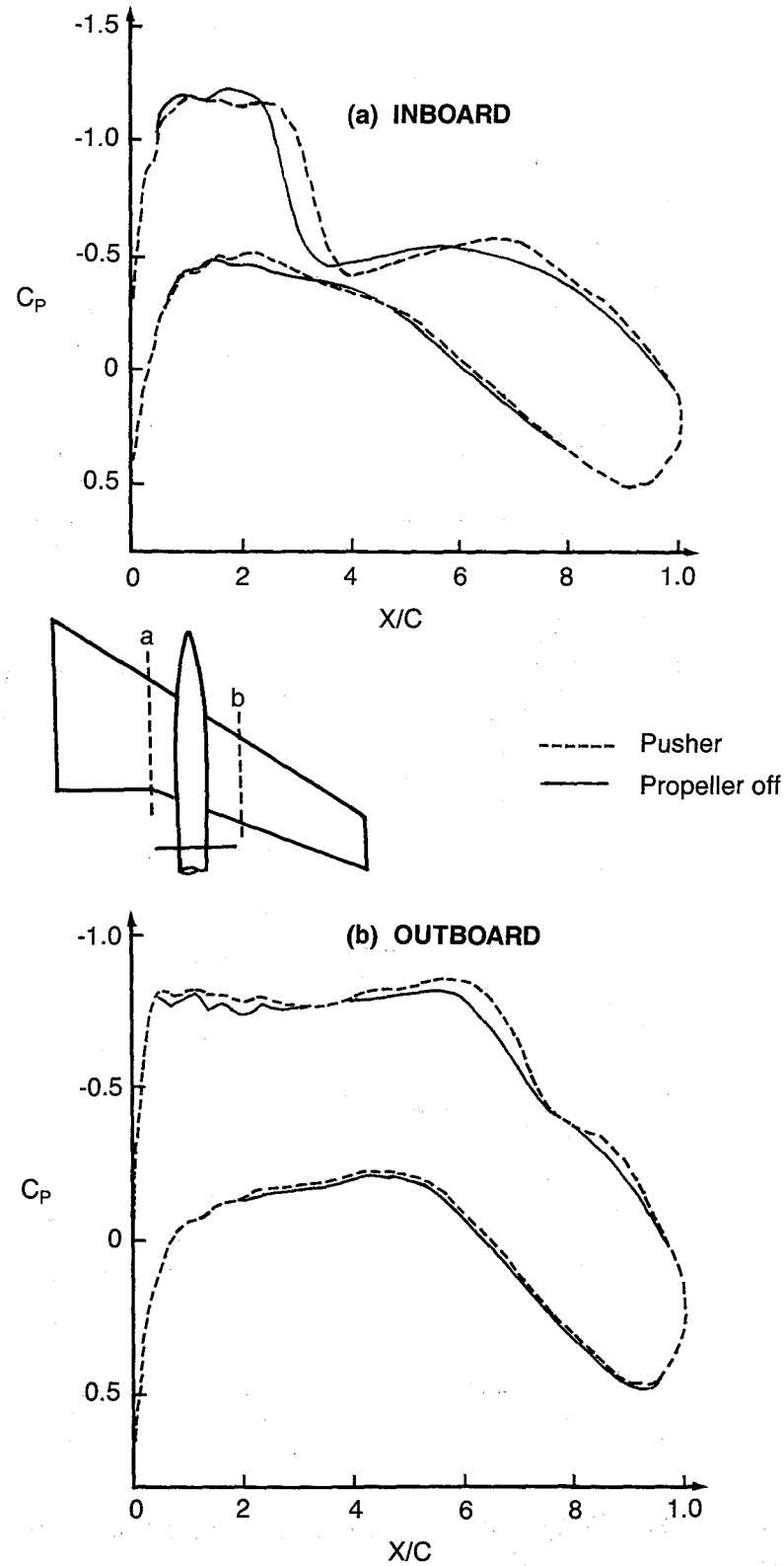
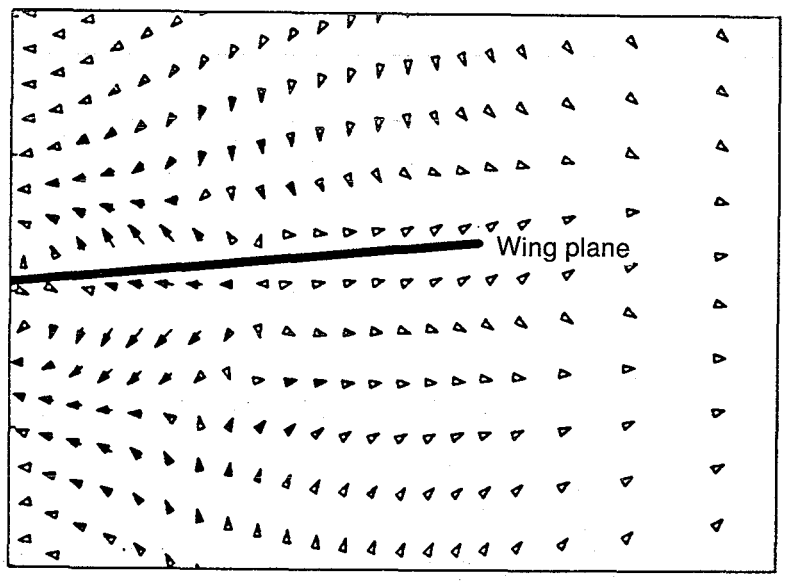


Figure 16. Comparisons of Wing Surface Pressures for a Wing-Mounted Pusher With Propeller-Off Analysis at $M_\infty = .80$, $\alpha = 1.84$

(a) TRACTOR



(b) PUSHER

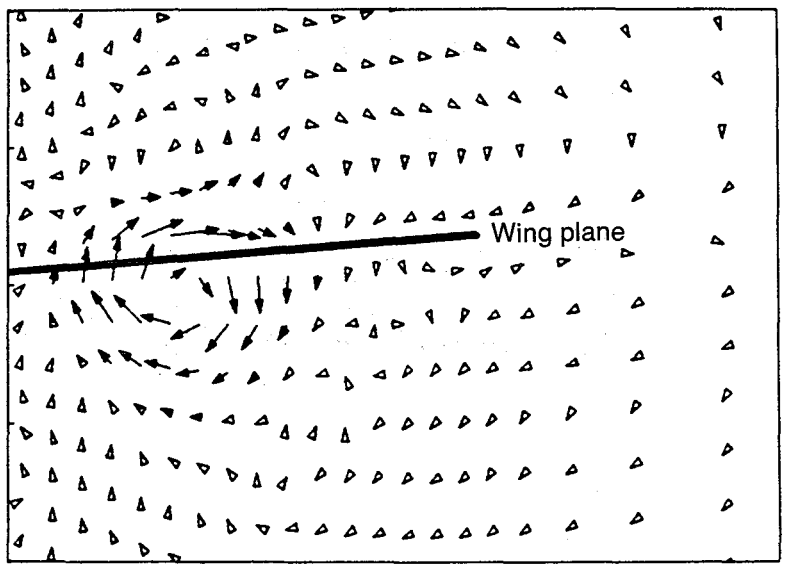


Figure 17. Comparisons of Velocity Difference Vector Plots for a Wing-Mounted Propfan at 2 Semispan Downstream of Wing Trailing Edge

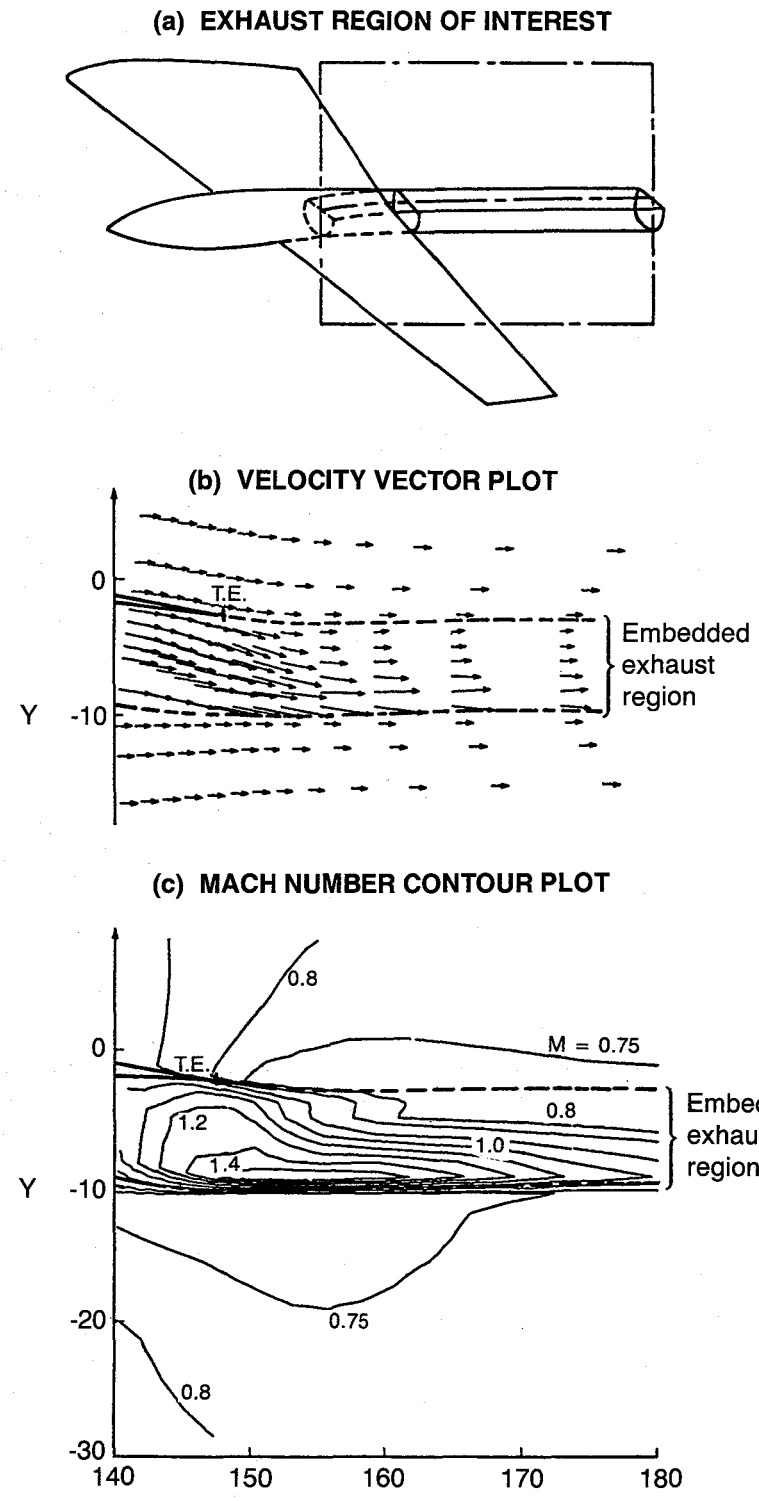


Figure 18. Exhaust Flow Grid Embedding for Turboprop Analysis Exhaust Conditions: $P_t/P_\infty = 2.6$ $T_t/T_\infty = 2.827$

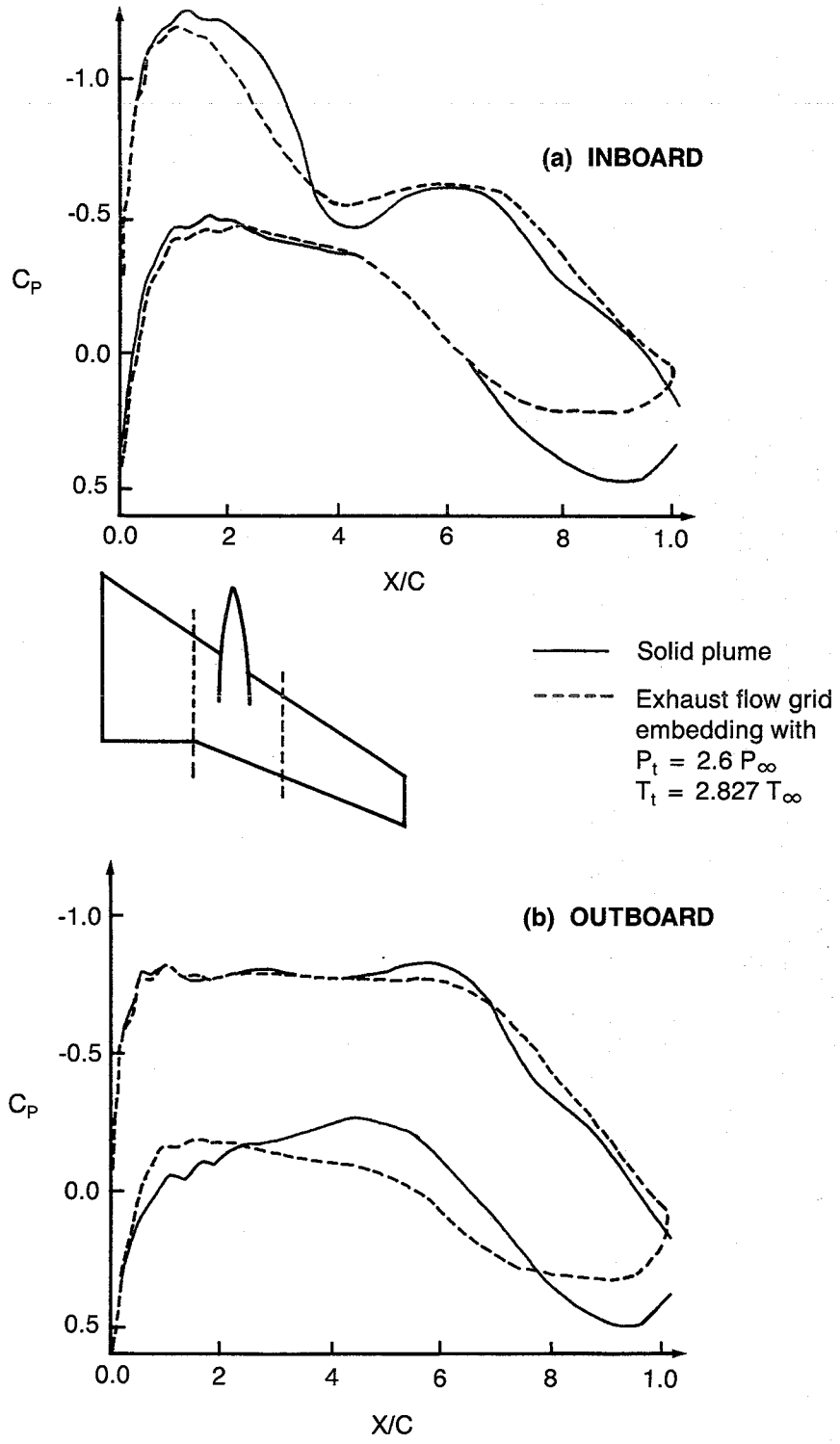


Figure 19. Effects of Exhaust Plume on Wing Surface Pressures

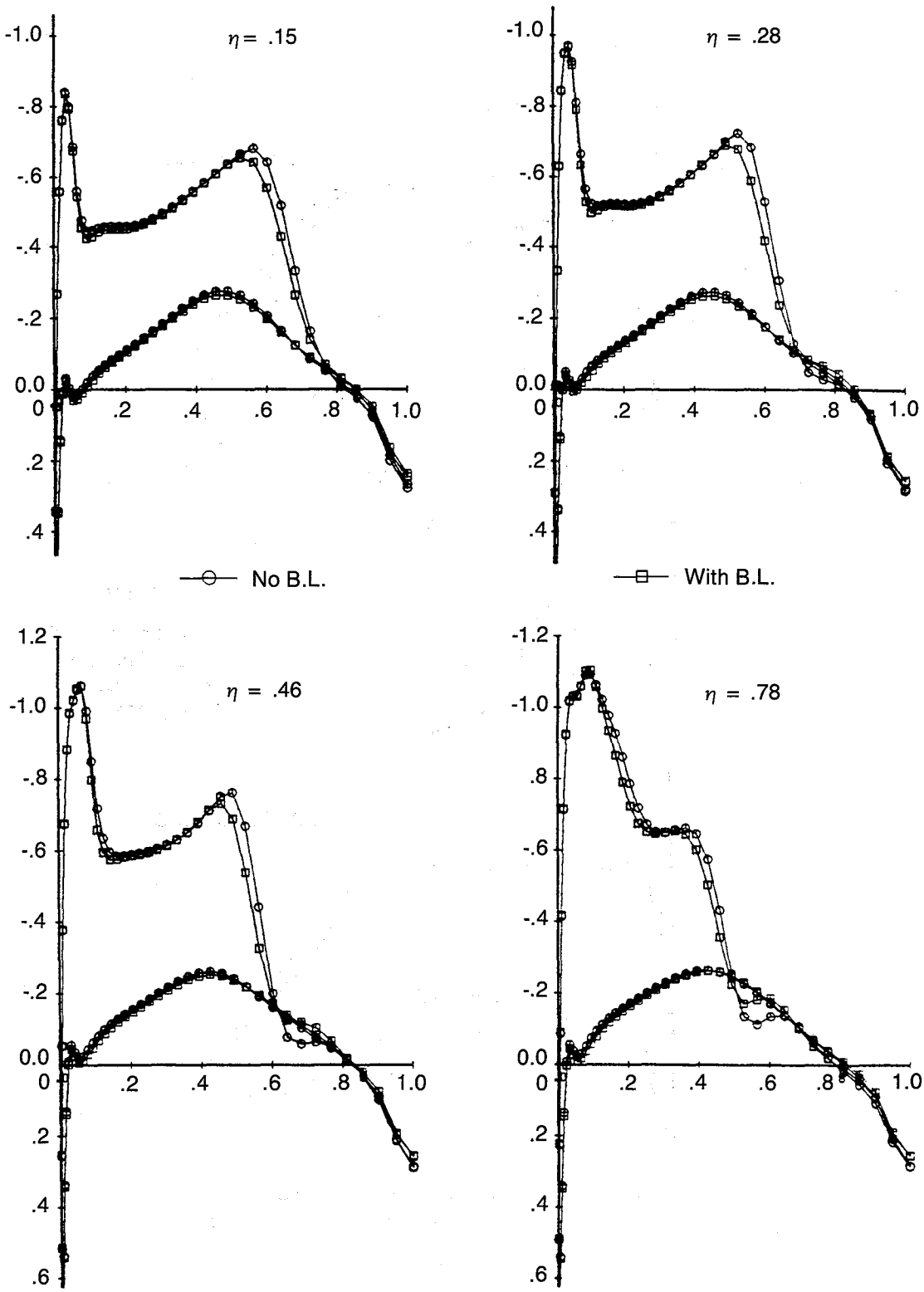


Figure 20. ONERA M6 Wing Analysis With and Without B.L. Coupling M = .84, α = 3.06°

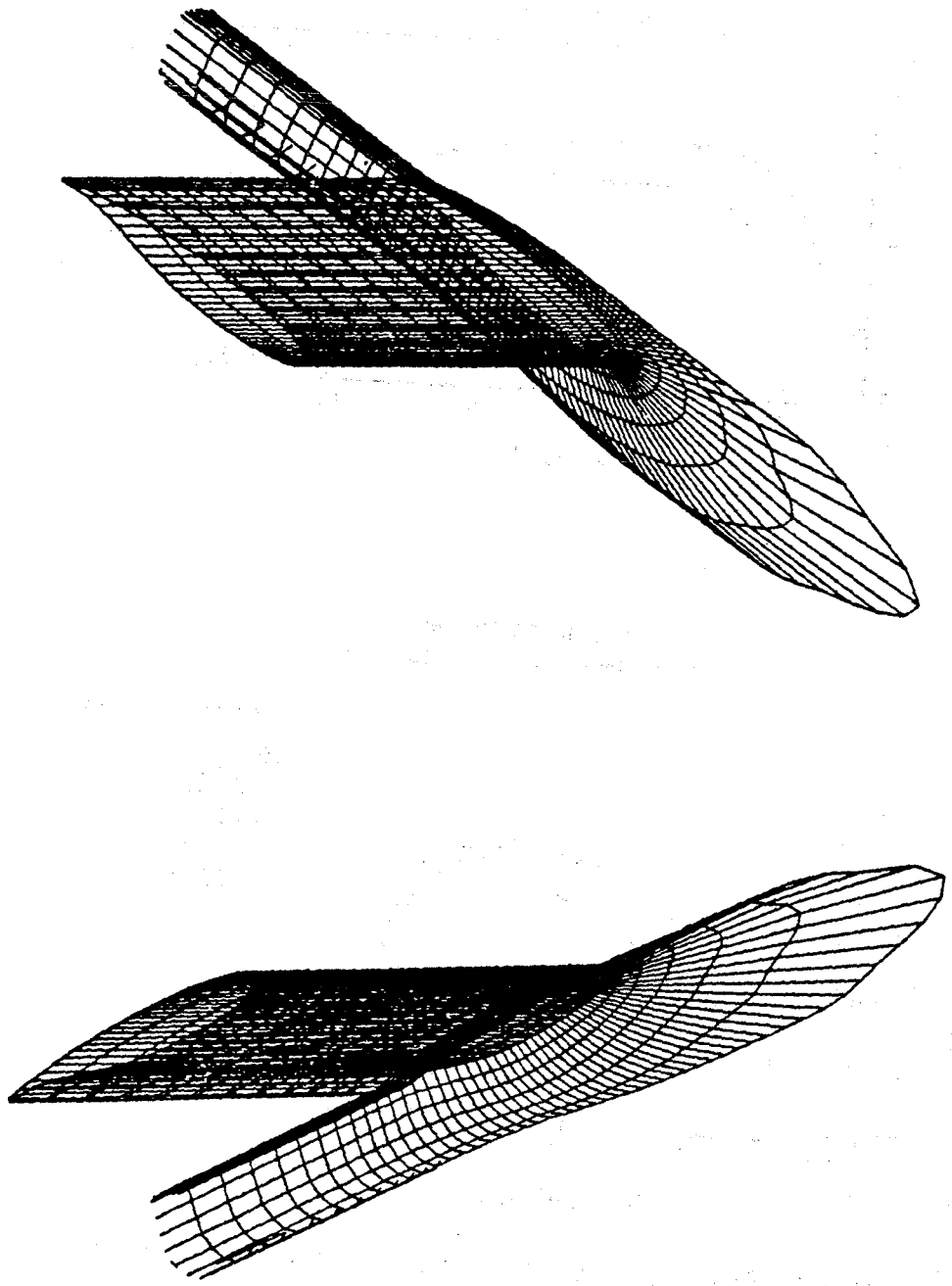


Figure 21. Typical Grid for NASA-Langley Wing-Nacelle Model (128 x 16 x 24)

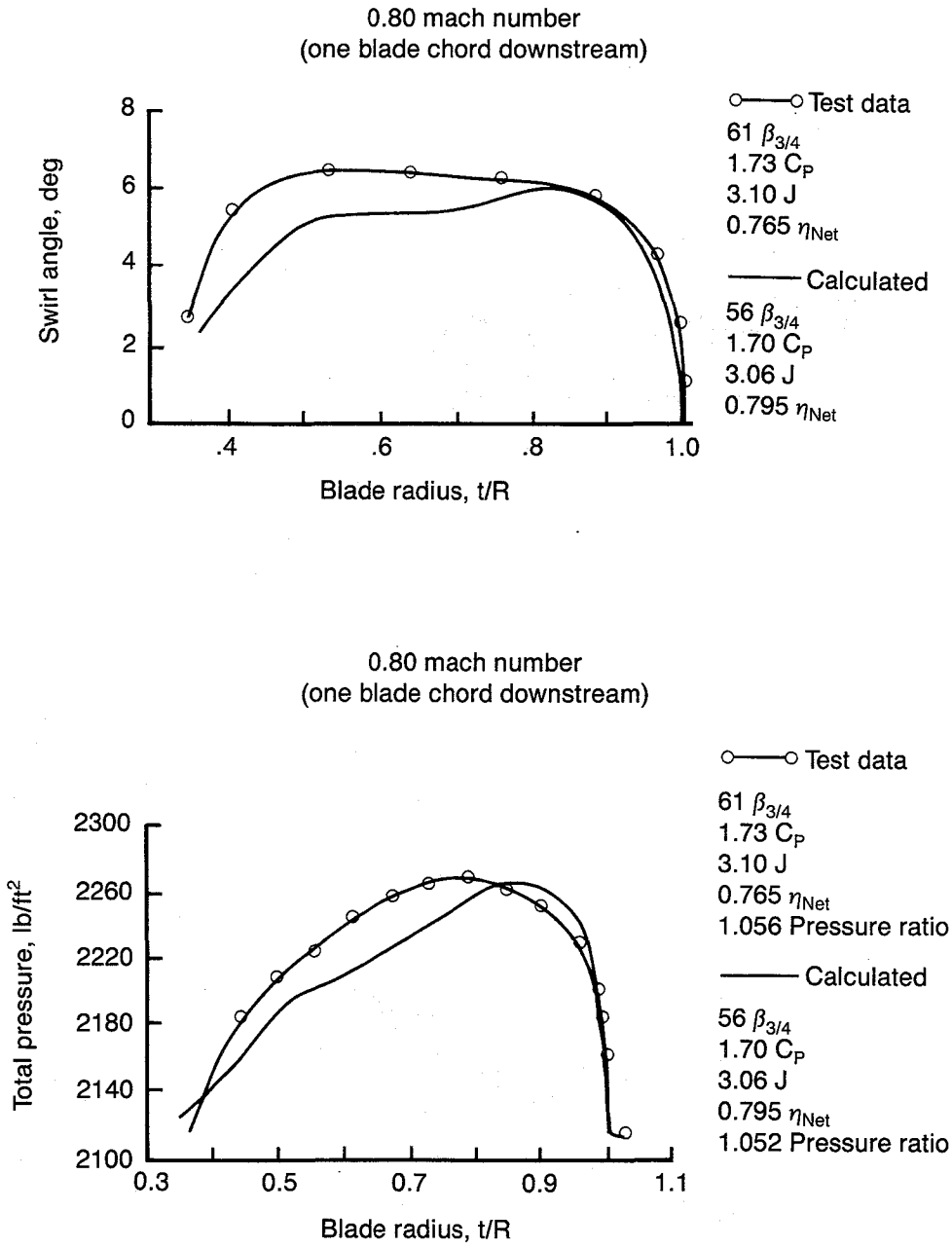
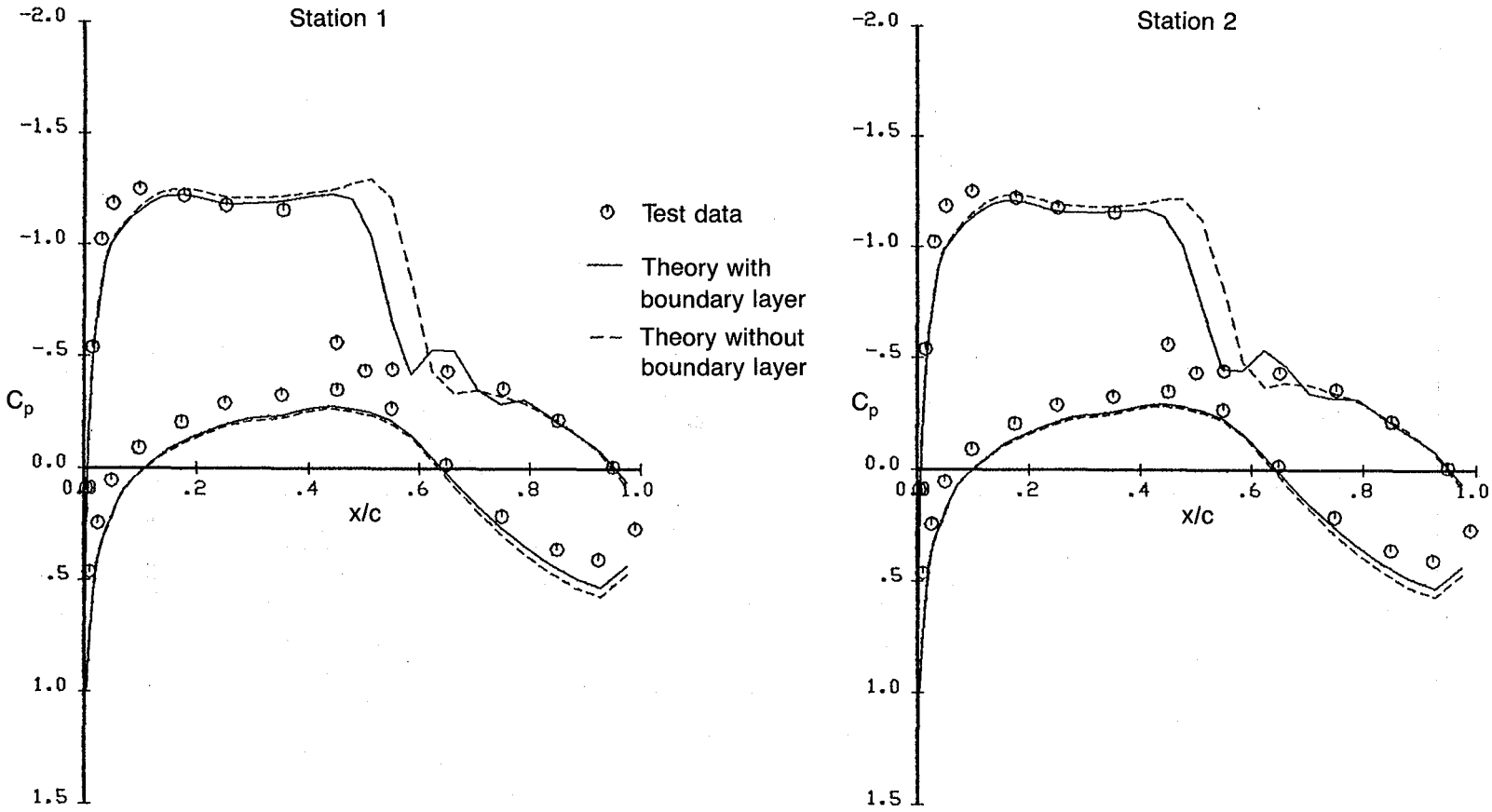
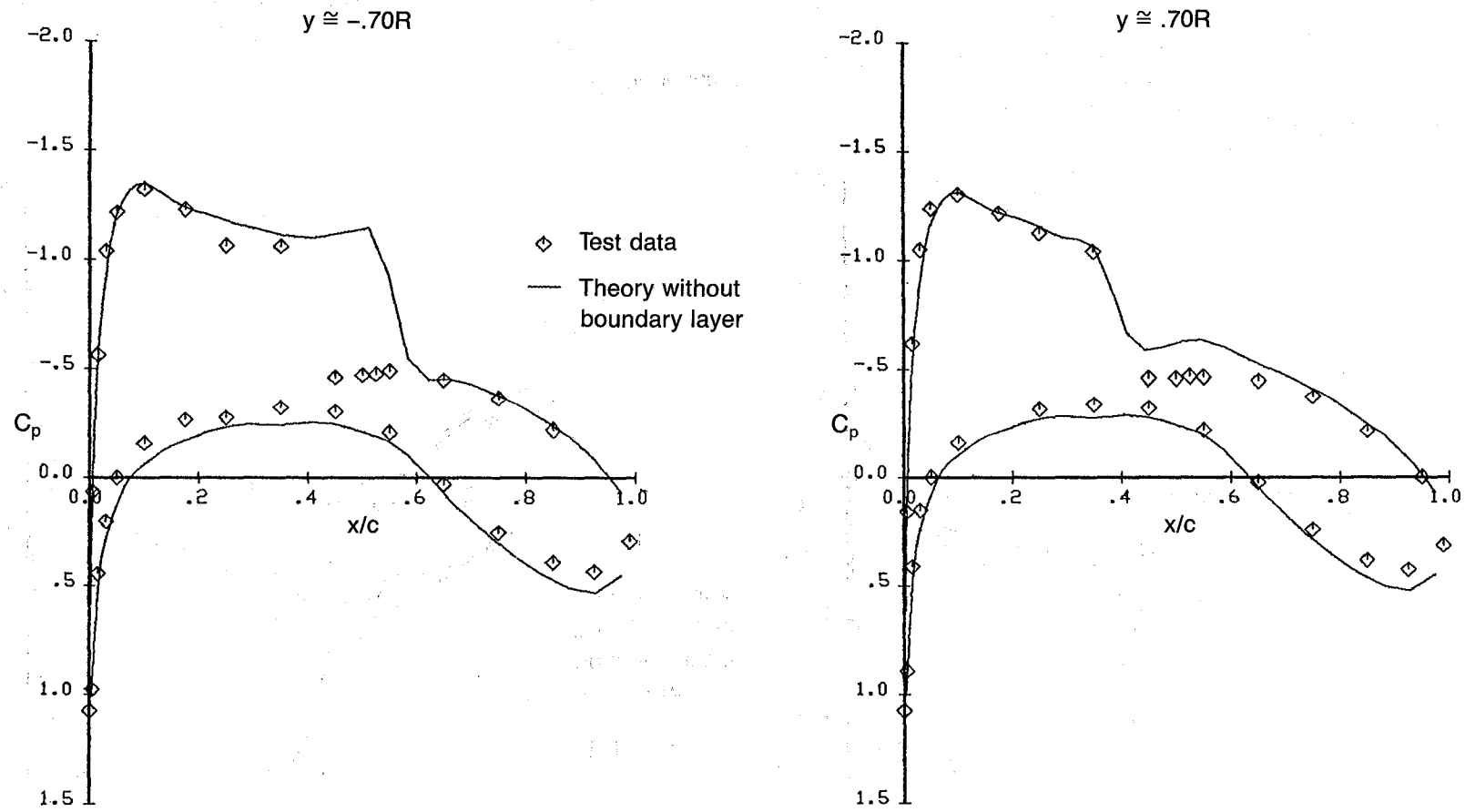


Figure 22. Propeller Loading Used in the Analysis of Wing-Nacelle Model



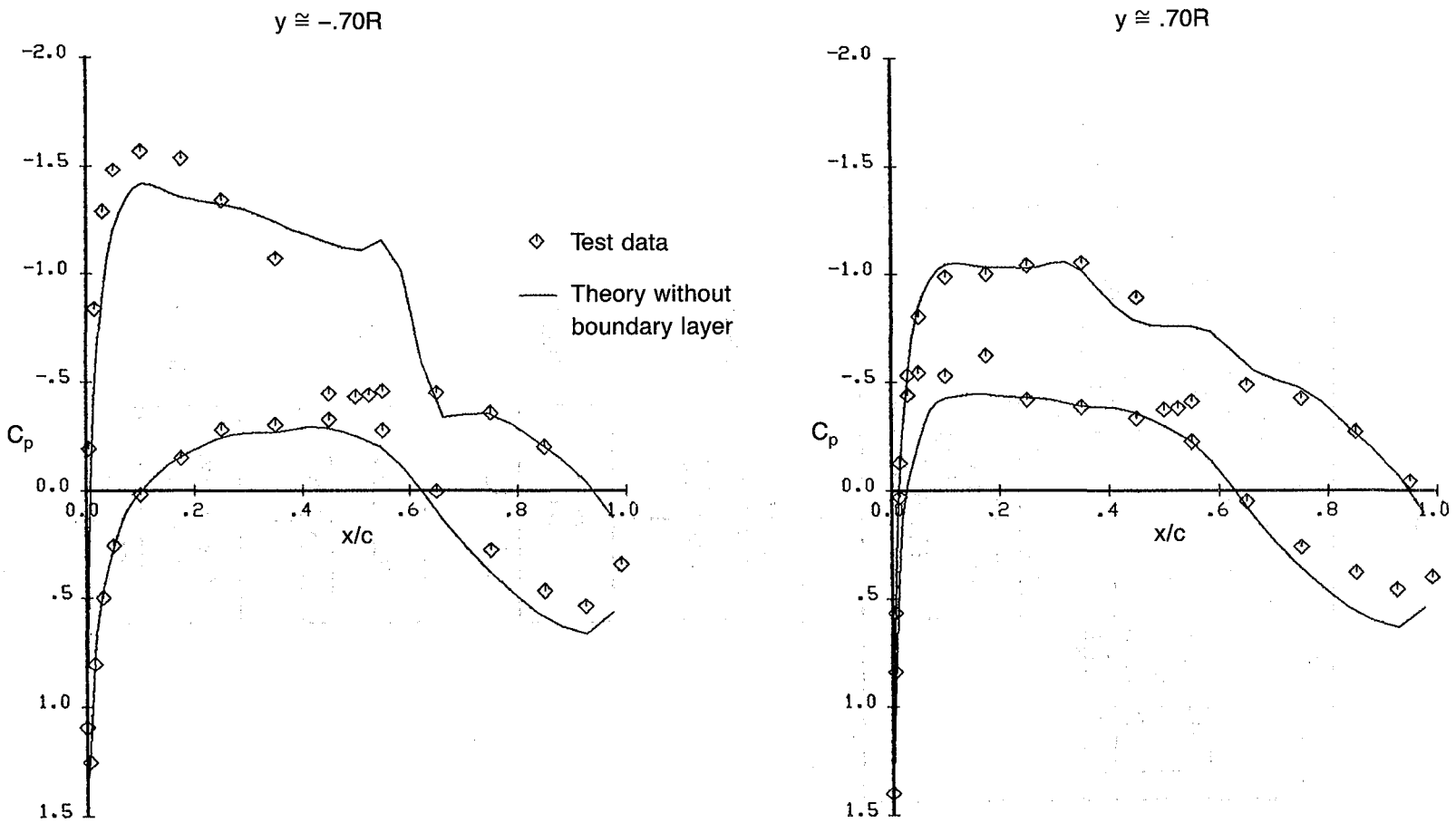
a) Wing Alone

Figure 23. The Test-Theory Comparison for NASA-Langley Wing-Nacelle Model. $M = .75$, $\alpha = 3.0$ deg



b) Wing-Nacelle

Figure 23. The Test-Theory Comparison for NASA-Langley Wing-Nacelle Model. $M = .75$, $\alpha = 3.0$ deg (Continued)



c) Wing-Nacelle-Prop

Figure 23. The Test-Theory Comparison for NASA-Langley Wing-Nacelle Model. $M = .75$, $\alpha = 3.0$ deg (Continued)

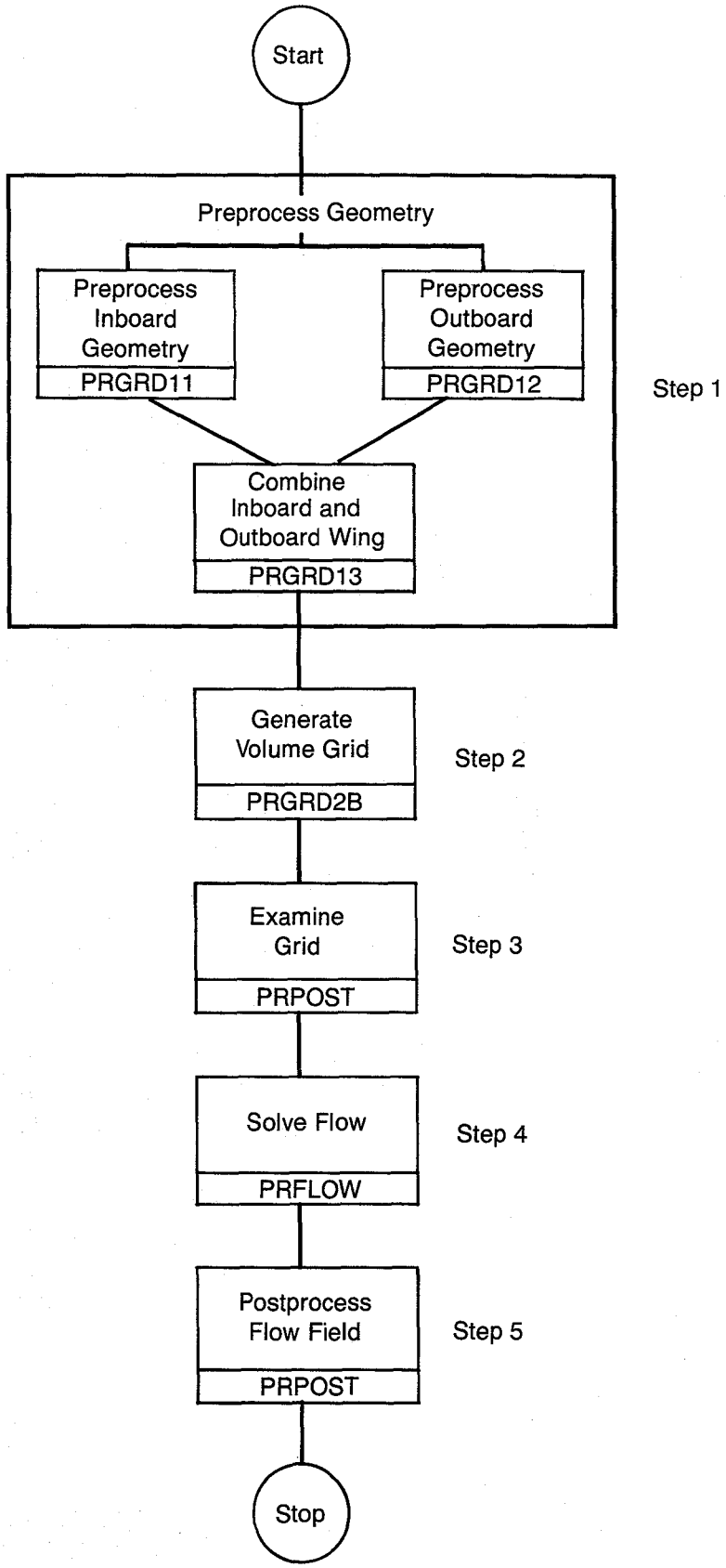


Figure A1. Flow Chart for System Execution

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-3954		

4. Title and Subtitle	5. Report Date
FLOW PREDICTION FOR PROPFAN ENGINE INSTALLATION EFFECTS ON TRANSPORT AIRCRAFT AT TRANSONIC SPEEDS	January 1986
	6. Performing Organization Code
7. Author(s) S. S. Samant and N. J. Yu	8. Performing Organization Report No.
9. Performing Organization Name and Address Boeing Commercial Airplane Company P.O. Box 3707 Seattle, WA 98124	10. Work Unit No.
	11. Contract or Grant No. NAS1-17250
	13. Type of Report and Period Covered Contractor Report
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546	14. Sponsoring Agency Code 505-45-43-05

15. Supplementary Notes
Langley Technical Monitor: John R. Carlson
Final Report

16. Abstract

An Euler-based method for aerodynamic analysis of turboprop transport aircraft at transonic speeds has been developed. In this method, inviscid Euler equations are solved over surface-fitted grids constructed about aircraft configurations. Propeller effects are simulated by specifying sources of momentum and energy on an actuator disc located in place of the propeller. A stripwise boundary layer procedure is included to account for the viscous effects. A preliminary version of an approach to embed the exhaust plume within the global Euler solution has also been developed for more accurate treatment of the exhaust flow. The resulting system of programs is capable of handling wing-body-nacelle-propeller configurations. The propeller disks may be tractors or pushers and may represent single or counterrotation propellers. Results from analyses of three test cases of interest (a wing alone, a wing-body-nacelle model, and a wing-nacelle-endplate model) are presented. A user's manual for executing the system of computer programs with formats of various input files, sample job decks, and sample input files is provided in appendices.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Propfan, Transonic Flow, Euler Equations	Unclassified—Unlimited Subject Category 02

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	102	A06

For sale by the National Technical Information Service, Springfield, Virginia 22161

NASA-Langley, 1986

End of Document